



**HAL**  
open science

# Les communications multipoints dans les réseaux haut débit multimédia : Le multicast en environnement IP sur ATM

Olivier Fourmaux

► **To cite this version:**

Olivier Fourmaux. Les communications multipoints dans les réseaux haut débit multimédia : Le multicast en environnement IP sur ATM. Réseaux et télécommunications [cs.NI]. Université Pierre et Marie Curie - Paris VI, 1998. Français. NNT : . tel-00008925

**HAL Id: tel-00008925**

**<https://theses.hal.science/tel-00008925>**

Submitted on 1 Apr 2005

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**Thèse de Doctorat de l'Université Pierre et Marie CURIE**

Spécialité

**SYSTÈMES INFORMATIQUES**

présentée par

**M. Olivier FOURMAUX**

pour obtenir le titre de

**DOCTEUR de l'Université Pierre et Marie CURIE**

Les Communications Multipoints  
dans les Réseaux Haut Débit Multimédia :  
**Le Multicast en Environnement IP sur ATM**

Soutenue le 14 Décembre 1998 devant le jury composé de

<b>M. Michel DIAZ</b>	<b>Président</b>
<b>M. Ken CHEN</b>	<b>Rapporteur</b>
<b>M. Jean-Jacques PANSIOT</b>	<b>Rapporteur</b>
<b>M. Patrick COCQUET</b>	<b>Examineur</b>
<b>M. Walid DABBOUS</b>	<b>Examineur</b>
<b>M. Guy PUJOLLE</b>	<b>Examineur</b>
<b>M. Serge FDIDA</b>	<b>Directeur de thèse</b>



*À Louise*  
*À ma famille*



# Remerciements

Cette thèse, intitulée “Les Communications Multipoints dans les Réseaux Haut Débit Multimédia : Le Multicast en Environnement IP sur ATM”, a été soutenue à l’Université Pierre et Marie Curie (PARIS VI) devant le jury composé de M. Michel DIAZ, M. Ken CHEN, M. Jean-Jacques PANSIOT, M. Patrick COCQUET, M. Walid DABBOUS, M. Guy PUJOLLE et M. Serge FDIDA.

Je tiens à remercier les membres de ce jury :

M. Michel DIAZ, Directeur de Recherche du CNRS au LAAS à Toulouse, qui m’a fait l’honneur de présider ce jury de thèse.

M. Ken CHEN,

Professeur à l’Institut Galilé, Université de Villetaneuse (PARIS XIII),  
et M. Jean-Jacques PANSIOT,

Professeur à l’Université Louis Pasteur (STRASBOURG I), qui ont accepté d’être rapporteurs de ma thèse, pour l’intérêt qu’ils ont porté à ce travail.

M. Patrick COCQUET, Ingénieur chez Dassault Électronique,

M. Walid DABBOUS, Chercheur de l’INRIA à Sophia-Antipolis,

et M. Guy PUJOLLE, Professeur à l’Université de Versailles Saint-Quentin, qui ont eu l’amabilité de participer à ce jury.

M. Serge FDIDA, Professeur à PARIS VI, qui a dirigé mes travaux de recherche et m’a soutenu pendant ces trois années.

Je remercie également les membres de l’équipe Réseaux et Performances avec qui j’ai apprécié travailler pendant ces trois années, et plus particulièrement : Cristofe DELEUZE, co-bureau de depuis le DEA, qui m’a supporté plus que de raison ; Vincent ROCA, Maître de Conférence à PARIS VI avec qui j’ai skié ; et mes deux stagiaires, Julien CALM et Thomas DIETZ, respectivement en stage de DEA Systèmes Informatiques (PARIS VI) et de *Diplomarbeit* (UNIVERSITÄT KARLSRUHE).



# Résumé

L'évolution des réseaux actuels vise principalement l'amélioration des performances à travers une augmentation importante des débits disponibles. Cependant, les besoins des applications multimédia se situent également dans l'accès à de nouvelles fonctionnalités telles que le multipoint et la garantie de qualité de service (QoS). Pour satisfaire ces besoins, nous proposons d'associer une architecture pour le multipoint avec QoS aux technologies de commutation.

Une première instanciation de cette approche nous a amenés à étudier l'intégration de RSVP sur les réseaux ATM, et à proposer une solution palliant l'absence de service multipoint avec QoS capable de soutenir efficacement RSVP. Nous nous intéressons ainsi à CLIP, l'une des techniques d'intégration d'IP sur ATM, pour réaliser l'intégration du modèle RSVP. La contribution repose sur l'utilisation de plusieurs arbres multipoints pour éviter les limitations intrinsèques aux réseaux ATM. Cette approche a été expérimentée sur la plate-forme MIRIHADÉ à l'aide d'une application vidéo à codage hiérarchique.

La commutation associée à une signalisation orientée connexion se prêtant mal à l'intégration des nouvelles fonctionnalités, nous nous sommes orientés vers l'utilisation d'une signalisation en mode non connecté à travers une solution de type "Commutation par label". Appliquée à RSVP, nous avons intégré directement la signalisation de la commutation dans celle assurant les nouvelles fonctionnalités pour permettre la commutation directe du trafic de données tout en conservant les fonctionnalités de la couche réseau. Notre solution, appelée "RSVP Switching", est en cours d'expérimentation sur la plate-forme SAFIR, pour un projet de simulation interactive distribuée ayant des besoins importants en terme de multipoint avec QoS.

## Mots-Clés :

Multipoint, Qualité de service, Réseaux haut débit, Multimédia, IP, Multicast, RSVP, ATM, Commutation IP.





# *Abstract*

New multimedia applications require more than a significant bandwidth increase. They need new functionalities from the network, such as multipoint communications and Quality of Service (QoS) guarantees. To satisfy those requirements, we suggest building an architecture enabling both multipoint and QoS on top of switching technologies.

As an application of integrating new functionalities with switching technologies, we suggest operating RSVP over ATM Networks. Several approaches to run IP over ATM are currently used, but none of them provide an efficient multipoint service with QoS to manage the full RSVP model. In a first contribution, we focus on a CLIP solution and extend it. The RSVP model is fully provided by our solution while preserving dynamic and heterogeneous QoS. It relies upon multiple multipoint trees to avoid intrinsic ATM model limitations. This solution has been implemented in the MIRIHADÉ testbed.

The use of switching technologies with connection-oriented signalling protocols is inefficient in providing new functionalities. Therefore, we propose using an hop-by-hop signalling protocol to cope with our needs. A new approach, named IP Switching, closely integrates routing and switching, and provides local ATM cut-through while using only IP signalling. In a second contribution, we improve this approach by introducing RSVP/IntServ. This solution, named “RSVP Switching”, is currently under experimentation in the SAFIR testbed for a distributed interactive simulation project with stringent QoS requirements.

## **Key Words:**

Multipoint, Quality of service, High-speed networks, Multimedia, IP, Multicast, RSVP, ATM, IP Switching.



# *Table des matières*

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Les services multipoints dans IP et ATM</b>	<b>7</b>
2.1	Introduction du chapitre . . . . .	8
2.2	Les services du monde Internet . . . . .	9
2.2.1	Services IP de base . . . . .	9
2.2.2	Services IP avec Multicast . . . . .	11
2.2.3	Services IP version 6 . . . . .	13
2.2.4	Services IP avec QoS . . . . .	15
2.3	Les services des réseaux ATM . . . . .	19
2.3.1	Principes de base d'ATM . . . . .	19
2.3.2	Services offerts à travers UNI . . . . .	22
2.3.3	Services ATM et multipoint . . . . .	26
2.4	Conclusion du chapitre . . . . .	29
<b>3</b>	<b>L'intégration d'IP sur les réseaux ATM</b>	<b>31</b>
3.1	Introduction du chapitre . . . . .	32
3.2	Les solutions classiques de l'IETF . . . . .	32
3.2.1	Classical IP and ARP over ATM (CLIP) . . . . .	33
3.2.2	Multicast Address Resolution Server (MARS) . . . . .	35
3.2.3	Next Hop Resolution Protocol (NHRP) . . . . .	41
3.2.4	IPv6 over ATM . . . . .	42
3.2.5	Perspectives . . . . .	43

---

3.3	Les solutions de l'ATM Forum . . . . .	43
3.3.1	LAN Emulation . . . . .	43
3.3.2	MultiProtocol Over ATM . . . . .	47
3.3.3	Perspectives . . . . .	51
3.4	Les solutions basées sur la commutation IP . . . . .	51
3.4.1	Approche orientée flot . . . . .	54
3.4.2	Approche orientée topologie . . . . .	58
3.4.3	Comparaison des <i>multicast</i> . . . . .	63
3.4.4	Perspectives . . . . .	66
3.5	Conclusion du chapitre . . . . .	66
<b>4</b>	<b>Multipoints avec "Classical IP"</b>	<b>69</b>
4.1	Introduction du chapitre . . . . .	70
4.2	L'approche RSVP sur <i>Classical IP over ATM</i> . . . . .	70
4.2.1	L'architecture RSVP pour les réseaux ATM . . . . .	71
4.2.2	Modèle d'intégration de RSVP sur les réseaux ATM . . . . .	72
4.2.3	Les problèmes liés à l'adaptation de RSVP sur ATM . . . . .	78
4.3	Étude de l'intégration du multicast . . . . .	80
4.3.1	Granularité de service multicast . . . . .	81
4.3.2	Mise en œuvre du modèle à arbres multiples . . . . .	85
4.4	Comparaison avec les propositions alternatives . . . . .	89
4.4.1	Approche RSVP sur ATM de l'IETF . . . . .	89
4.4.2	Modèle avec liaisons ATM directes . . . . .	90
4.4.3	Entités de coordination RSVP . . . . .	92
4.4.4	MIS et EARTH . . . . .	93
4.5	L'expérimentation . . . . .	94
4.5.1	Système GNU/Linux ATM . . . . .	94
4.5.2	La plate-forme MIRIHADÉ . . . . .	96
4.5.3	Application Vidéo . . . . .	99
4.5.4	Les résultats . . . . .	103
4.6	Conclusion du chapitre . . . . .	104

---

<b>5</b>	<b>Multipoints avec la Commutation par Label</b>	<b>107</b>
5.1	Introduction du chapitre . . . . .	108
5.2	La proposition RSVP Switching . . . . .	109
5.2.1	Le choix d'une approche . . . . .	109
5.2.2	IP Switching et QoS . . . . .	110
5.2.3	Notre proposition . . . . .	110
5.3	L'intégration du service multipoint de RSVP . . . . .	112
5.3.1	Cut-through . . . . .	112
5.3.2	Filtrage . . . . .	115
5.3.3	Agrégation . . . . .	118
5.4	Spécialisation du modèle RSVP Switching . . . . .	120
5.4.1	IPv6 et RSVP Switching . . . . .	121
5.4.2	ATM et RSVP Switching . . . . .	122
5.5	Comparaison avec d'autres intégrations . . . . .	124
5.5.1	IS-IPSO et QoS . . . . .	124
5.5.2	MPLS et QoS . . . . .	124
5.5.3	QSR et QoS . . . . .	125
5.6	Modification de RSVP . . . . .	125
5.6.1	Ajout de la gestion des labels dans RSVP . . . . .	125
5.6.2	Souche RSVP de base . . . . .	127
5.6.3	Modification de l'interface avec le noyau . . . . .	128
5.7	Réalisation d'un RSVP Switch . . . . .	129
5.7.1	Plate-forme RSVP Switching . . . . .	130
5.7.2	Émulation d'un RSVP Switch . . . . .	130
5.7.3	Intégration de Service et RSVP Switching . . . . .	133
5.8	L'expérimentation . . . . .	136
5.8.1	Les applications DIS . . . . .	136
5.8.2	Intégration du DIS avec RSVP . . . . .	138
5.8.3	Intégration du DIS avec RSVP Switching . . . . .	138
5.8.4	Observations . . . . .	139

5.9 Conclusion du chapitre . . . . .	140
<b>6 Conclusion</b>	<b>141</b>
<b>Publications</b>	<b>145</b>
<b>Bibliographie</b>	<b>147</b>
<b>Liste des acronymes</b>	<b>157</b>
<b>Liste des figures</b>	<b>161</b>
<b>Liste des tableaux</b>	<b>165</b>

# CHAPITRE 1

## *Introduction*

### LE MULTIPPOINT AVEC QOS DANS LES RÉSEAUX HAUT DÉBIT

La croissance exponentielle des besoins en terme de réseaux implique, en plus d'une augmentation considérable du débit, l'intégration de nouvelles fonctionnalités. Celles-ci sont nécessaires d'une part pour utiliser de manière efficace les ressources disponibles et d'autre part pour satisfaire les récents besoins des applications multimédia. Les fonctionnalités auxquelles nous nous intéressons sont associées aux services multipoints avec garanties de Qualité de Service (QoS).

Les réseaux actuels ne fournissent généralement qu'un service "au mieux" point-à-point, c'est-à-dire d'un émetteur vers un récepteur sans aucune garantie. Un service aussi restreint n'est pas apte à satisfaire les nouveaux besoins, tels que ceux liés au transport de flux vidéo entre plusieurs émetteurs et plusieurs récepteurs. Les applications ont alors besoin d'un service multipoint pour permettre des transmissions efficaces entre plusieurs entités. Conjointement au multipoint, la QoS est inévitable pour répondre aux contraintes des nouveaux trafics. Divers paramètres entrent en jeu dans la notion de QoS, tels le débit, le délai, la fiabilité ou la synchronisation. Nous nous limiterons aux caractéristiques généralement associées à la couche réseau, c'est-à-dire le débit et le délai.

L'augmentation des débits à supporter implique l'utilisation de techniques de plus en plus performantes telles que la commutation. Associée à celle-ci, une signalisation spécifique permet la coordination des différents éléments commutants. La principale technologie de commutation actuelle est ATM (*Asynchronous Transfer*



*Mode*), et permet l'accès à des performances élevées. ATM est généralement utilisé avec la signalisation orientée connexion de l'ATM Forum ou de l'ITU-T. Des approches récentes proposent de recourir à d'autres signalisations basées sur le mode non connecté. L'utilisation de services avec multipoint et QoS implique l'emploi d'une signalisation associée à ces services. Par exemple au sein de l'Internet, des extensions à IP (*Internet Protocol*) telles que le multicast et RSVP (*Resource Reservation Protocol*) ont été développées.

Pour atteindre notre but, qui est de fournir des services de multipoint avec QoS et des performances élevées, l'intégration d'une signalisation pour les nouveaux services et d'une signalisation de commutation est nécessaire. C'est ce que nous proposons de faire à travers les contributions de cette thèse. L'originalité de ce travail réside dans la proposition de solutions intégrant le multipoint et la QoS avec le haut débit, contrairement à la tendance générale qui vise principalement à améliorer les performances des services existants sans associer de nouvelles fonctionnalités.

## LES SERVICES DANS INTERNET ET LES RÉSEAUX ATM

Dans le but de proposer des solutions intégrant le multipoint avec QoS dans des réseaux haut débit, nous nous intéressons aux deux technologies majeures que sont IP, le protocole de base de l'Internet, et ATM, la technologie sous-jacente aux réseaux large bande à intégration de services (RNIS-LB). Ces deux technologies sont souvent mises en opposition par leur recours à des techniques différentes et souvent orthogonales, ainsi que par leur appartenance à deux mondes différents : celui des informaticiens pour IP, et celui des télécommunications pour ATM. Nous proposons dans ce document une utilisation conjointe de ces deux solutions pour servir de support à nos propositions.

Outre leurs succès respectifs, nous nous sommes intéressés à IP et à ATM pour leurs caractéristiques fondamentales : IP est le protocole d'accès au réseau dominant et sera amené à se généraliser plus encore ; ATM est une technologie très haut débit. Nous étudions dans ce cas les possibilités d'interfonctionnement d'IP et d'ATM, et plus particulièrement d'IP **sur** ATM, de manière à associer le meilleur des deux mondes : la souplesse d'IP et les performances d'ATM.

L'utilisation d'IP sur ATM est un sujet très vaste, ouvrant sur de nombreux champs de recherche allant des aspects signalisation, avec la proposition de modèles pour la mise en correspondance des services des deux technologies, jusqu'aux études de performances des différents modèles d'intégration proposés. Pour notre part, nous nous sommes restreints à un type de service particulier : celui permettant des communications entre un nombre variable d'entités, appelé communication multipoint, avec des garanties de qualité de service. Cette thèse présente donc, comme instantiation de la problématique des nouvelles fonctionnalités en environnement haut débit, l'intégration des services multipoints avec QoS dans le cadre d'IP sur ATM.

## Le service Internet

L'Internet offre un service réseau point-à-point non connecté (*unicast*) sans garanties (*Best Effort*). Ce service est suffisant pour beaucoup d'applications, mais certaines l'utilisent par défaut d'autres possibilités. L'absence de services avec multipoint et QoS limite l'apparition de nouvelles applications utilisant des média tels que la vidéo ou l'audio. Des extensions à IP ont été développées pour pallier ces lacunes. La première est **IP multicast** et introduit le service multipoint avec la notion d'abstraction de groupe. Pour la QoS, le protocole de signalisation **RSVP** et les classes de services de l'IntServ permettent l'introduction de garanties de QoS sur un modèle multipoint-à-multipoint avec des valeurs de QoS renégociables et différentes suivant les destinataires. Ces extensions sont encore en phase expérimentale et de nombreux problèmes tels que la résistance au facteur d'échelle (*Scalability*) demeurent. La dernière version du protocole IP, **IPv6**, intègre dès à présent le multicast, mais il est pour l'heure aussi en phase expérimentale.

Le service multipoint avec QoS, fourni par IP multicast et RSVP, est celui disponible au niveau des applications. Il repose sur un modèle étendu du multipoint. C'est le service que nous souhaitons intégrer.

## Le service ATM

La technologie ATM a été développée pour répondre aux besoins supposés des futures applications. Les réseaux ATM fournissent, grâce à la commutation de cellules, des performances élevées avec l'accès à de très hauts débits et des délais de latence réduits. Outre des performances élevées, ATM fournit des facilités limitées pour les services multipoints avec QoS. Deux types d'utilisation des réseaux ATM sont envisageables :

- Le premier type utilise la signalisation orientée connexion habituellement associée aux réseaux ATM. Le service est basé sur des connexions point-à-multipoint avec une QoS homogène et statique vers chaque destinataire. Le service est malheureusement réduit par la spécificité de l'interface utilisée, impliquant l'utilisation de couches d'adaptation pour intégrer d'autres protocoles. Cette adaptation masque généralement les possibilités des réseaux ATM, tels que le multipoint et la QoS.
- Le deuxième type évite le mode connecté grâce à l'utilisation de nouvelles signalisations non connectées. Cette approche récente permet d'accéder de manière simplifiée aux performances de la commutation.

## L'intégration d'IP sur les réseaux ATM

Les travaux visant l'intégration d'IP avec ATM ont pour but de fournir des performances élevées sans considérer le multipoint et la QoS dans un premier temps. Cette

intégration limitée est déjà loin d'être évidente en raison des approches orthogonales des deux technologies (en particulier pour l'adaptation du mode non connecté d'IP sur le mode connecté des réseaux ATM). Les premières solutions ont conservé la signalisation orientée connexion d'ATM. Des groupes de travail de l'IETF ont produit *Classical IP* (CLIP), mais celui-ci ne propose ni multipoint, ni QoS, rendant nécessaire l'adjonction de nombreuses extensions (NHRP, MARS, RSVP...). *LAN Emulation* (LANE) ainsi que son extension au niveau réseau *MultiProtocol Over ATM* (MPOA) sont proposées au sein de l'ATM Forum, mais ne supportent pas non plus les services que nous attendons. Les solutions les plus récentes proposent l'utilisation de signalisation non connectée, telle que la commutation IP, et permettent une intégration plus aisée du multipoint avec QoS dans les réseaux haut débit.

## L'INTÉGRATION DES NOUVELLES FONCTIONNALITÉS SUR LES RÉSEAUX COMMUTÉS

Dans le cadre de l'intégration des nouvelles fonctionnalités sur les réseaux à hautes performances, nous avons développé deux propositions. La première s'appuie sur l'intégration d'une signalisation pour le multipoint avec QoS sur un service réseau orienté connexion, ce qui se matérialise en l'utilisation de RSVP sur les réseaux ATM usuels. La seconde associe la signalisation pour les nouvelles fonctionnalités avec un service réseau non connecté et correspond à l'utilisation de RSVP avec la commutation IP.

### L'intégration en environnement connecté

Sur la base de l'absence de proposition de service multipoint avec QoS, nous avons développé une solution basée sur CLIP et ayant pour but l'intégration complète du service RSVP. Cette approche est l'une des principales pour accéder aux services avec QoS des réseaux ATM à travers la couche IP. Cependant, même si l'intégration de RSVP dans CLIP a le soutien de la communauté Internet à travers l'IETF, les solutions proposées sont souvent incomplètes et ne gèrent qu'une partie du modèle multipoint présent dans RSVP. Nous proposons, à travers notre solution basée sur de multiples arbres point-à-multipoint, un modèle qui intègre l'ensemble des fonctionnalités de RSVP tout en conservant les caractéristiques majeures associées au service multicast avec QoS d'IP. Celles-ci sont l'hétérogénéité au niveau de la QoS des récepteurs dans une même session et la dynamique avec les aspects de renégociation et de gestion d'un groupe qui évolue dans le temps. Notre proposition présente surtout les aspects gestion des VC, ce qui concerne la mise en correspondance entre les paramètres de QoS d'ATM et de IP étant déjà largement traité.

Cette première proposition a donné lieu à la réalisation d'un prototype appliqué à la diffusion vidéo avec codage hiérarchique. Lors des expérimentations, la durée de

l'établissement des connexions s'est révélée considérable et incompatible avec les modèles de communication multipoint dans l'état actuel de la signalisation ATM pour les grandes distances. Cet aspect est particulièrement important pour IP et RSVP, associés à un modèle dynamique impliquant des créations fréquentes de connexions ATM.

### **L'intégration en environnement non connecté**

Dans un deuxième temps nous proposons l'intégration d'une signalisation pour le multipoint avec QoS et d'une signalisation non connectée pour la commutation. La première signalisation est toujours RSVP. La signalisation pour la commutation est pour sa part basée sur la technique de "Commutation de Label". Dans ce cadre, nous développons notre solution "**RSVP Switching**" qui met en œuvre les mécanismes nécessaires au fonctionnement du modèle multipoint-à-multipoint hétérogène et dynamique du service IP multicast avec QoS dans un environnement commuté. Nous développons particulièrement l'adaptation des mécanismes d'agrégation et de filtrage de RSVP au niveau de la commutation. La spécialisation de notre solution à IPv6 et à la commutation ATM permet d'éviter le mode connecté des réseaux ATM. Nous obtenons ainsi une approche intégrant directement la signalisation IP sur la partie commutation d'ATM (mise en œuvre de commutateurs sans la signalisation usuelle). Ce type d'approche pour l'intégration d'IP sur ATM est appelé "Commutation IP" et a pour avantage, entre autres, de considérablement simplifier l'intégration des deux technologies tout en profitant au mieux de la performance liée à la commutation ATM et des avantages de la signalisation IP. Nous proposons ensuite une mise en œuvre au sein de notre plate-forme ATM longue distance et projetons une validation à l'aide d'applications de DIS aux besoins en QoS importants.

Ce travail s'appuie en particulier sur la plate-forme ATM locale développée au sein du LIP6. Cette plate-forme, liée à celle du campus de Jussieu, compose la base des successives plates-formes ATM nationales auxquelles nous avons participé et sur lesquelles nous avons mené des expérimentations : MIRIHADÉ puis SAFIR. Au niveau IP nous avons développé localement diverses plates-formes, que ce soit pour IP multicast, IPv6 ou RSVP, avant de fédérer ces technologies au sein du projet DIS/ATM et de pouvoir intégrer nos solutions.

### **PLAN DE LA THÈSE**

Le corps de la thèse est constitué de quatre chapitres. Le Chapitre 2 présente les services en général, et le multipoint avec QoS en particulier, des deux principales technologies de réseaux : Internet et ATM. Le Chapitre 3 décrit les différentes intégrations de ces deux technologies, qu'elles soient basées sur des approches usuelles avec CLIP et LANE, ou sur des approches récentes avec la commutation IP. Le Chapitre 4 propose une première contribution visant à assurer un service multipoint

avec QoS à l'aide de RSVP sur les réseau ATM. Suite à ces travaux, une intégration plus efficace est envisagée au Chapitre 5, utilisant l'approche de la commutation IP. Un dernier chapitre présente les conclusions de ces travaux et les perspectives.

## CHAPITRE 2

# *Les services multipoints dans IP et ATM*

### Contenu du chapitre

*Ce chapitre présente les services fournis par les deux architectures de réseaux majeures pour la mise en œuvre des services multipoints avec QoS : IP et ATM. Les services disponibles ainsi que ceux qui le seront dans un futur proche y sont décrits. Notre but n'est pas de détailler précisément les différentes caractéristiques offertes par ces technologies mais plutôt d'insister sur les aspects liés aux nouvelles fonctionnalités.*

*Ce chapitre est organisé de la manière suivante :*

- ⇒ Section 2.1 – Introduction du chapitre – p. 8 : *Présentation des deux environnements.*
- ⇒ Section 2.2 – Les services du monde Internet – p. 9 : *Description du service IP de base, du service IP Multicast, du service IPv6 et du service avec QoS à travers les travaux des groupes RSVP et IntServ.*
- ⇒ Section 2.3 – Les services des réseaux ATM – p. 19 : *Présentation des principes de base et des services des réseaux ATM, et en particulier de la QoS et du multipoint.*
- ⇒ Section 2.4 – Conclusion du chapitre – p. 29 : *Comparaison des services.*

## 2.1 INTRODUCTION DU CHAPITRE

Les réseaux actuels fournissent un service de base, généralement limité aux communications point-à-point sans QoS. Les nouvelles applications multimédia réclament un service étendu pouvant satisfaire leurs besoins tant au niveau du multipoint que de la QoS. Ces services correspondent à la mise en œuvre de communications entre plusieurs entités et ont de forts besoins en terme de nouvelles fonctionnalités (pour une étude plus approfondie de ces besoins, voir l'article [FFO97]). Nous abordons en conséquence le problème des services multipoints conjointement avec celui de la QoS, ces deux notions étant indissociables pour envisager des solutions pérennes.

Les communications multipoints suscitent un grand intérêt, de par les nombreuses applications envisagées et le manque de disponibilité des mécanismes nécessaires à leur mise en œuvre dans de nombreux média. Cependant, ce type de communication soulève des problèmes importants présentés dans l'article [DDC97]. Deux catégories sont à distinguer selon que l'on se situe en mode datagramme ou connecté :

- En mode datagramme, les communications multipoints sont qualifiées de *multicast*, terme dérivant de *broadcast* que l'on associe à la diffusion non sélective inhérente à certaines technologies de réseau. Le *multicast* est, quant à lui, associé à une diffusion sélective ou limitée vers un certain ensemble de récepteurs de cardinalité  $N$  avec  $0 \leq N < +\infty$  ( $N$  est supposé pouvoir être très grand, on s'appliquera donc à trouver des solutions s'adaptant au facteur d'échelle). Le *multicast* est opposé à l'*unicast* qui qualifie la transmission vers un récepteur. L'environnement non connecté n'associant pas un émetteur particulier, la transmission *unicast* sera qualifiée de N-vers-1 (*many-to-one*), et le *multicast* de N-vers-M (*many-to-many*).
- En mode connecté, la connexion simple liant un émetteur et un récepteur est appelée connexion "point-à-point". Pour les communications multipoints, on parlera de "point-à-multipoint" ou de "multipoint-à-multipoint" selon le nombre d'émetteurs impliqués. Le terme *multicast* est souvent associé de manière erronée au "point-à-multipoint" alors qu'il correspond en fait au "multipoint-à-multipoint".

Cette clarification est nécessaire car les deux architectures que nous allons traiter appartiennent évidemment chacune à un mode différent.

Notre présentation des services relatifs à IP et à ATM introduit deux environnements orthogonaux sur de nombreux points :

- L'environnement IP ou de l'**Internet** est principalement lié à l'industrie informatique. Celle-ci promeut un modèle où le réseau fournit un service minimaliste et où l'intelligence se trouve au sein des machines d'extrémités. Les organismes de standardisation associés sont principalement l'*Internet Engineering Task Force* (IETF) pour l'ingénierie et l'*Internet Research Task Force* (IRTF) pour la recherche. Ceux-ci coordonnent de nombreux groupes de travail qui

publient des documents préliminaires (*Internet Draft*) à validité limitée dans le temps, et des RFC (*Request For Comment*) qui ont valeur de **standards de fait**. Le maître mot pour l'adoption d'un standard est le pragmatisme et le large consensus : "*rough consensus and working codes*".

- L'environnement **ATM** est plutôt lié aux opérateurs de télécommunications et promeut un modèle où l'intelligence est préférentiellement située au sein du réseau. Les deux grands organismes associés sont l'Union Internationale des Télécommunications, secteur des Télécommunications (UIT-T ou *ITU-T* en anglais, anciennement appelé CCITT), et l'ATM Forum qui regroupe les principaux acteurs de cette technologie. L'UIT-T et l'ATM Forum publient des **standards de loi**.

## 2.2 LES SERVICES DU MONDE INTERNET

Le réseau d'interconnexion mondial, Internet, a pour protocole de base IP. Ce protocole s'est imposé, non seulement dans les environnements d'interconnexion, en tant que protocole de la couche réseau, mais aussi associé à des protocoles de niveau transport, en tant que protocole d'accès standard aux ressources communicantes. L'usage des services réseaux actuels passe donc principalement à travers ce protocole, qui peut être vu comme niveau de **convergence** pour les diverses technologies sous-jacentes. L'utilisation d'un protocole unique au dessus de réseaux de technologies variées est l'un des concepts principaux de l'architecture Internet.

Nous présentons dans cette section le service IP de base, puis son extension majeure pour les communications multipoints, appelée IP Multicast. Nous décrivons ensuite la nouvelle version de ce protocole, appelée IPv6 et terminons en présentant les services avec QoS associées à IP.

### 2.2.1 Services IP de base

Le protocole IP [Pe81] fonctionne en mode non connecté et offre un service dit "*Best Effort*" ou "Au Mieux", c'est-à-dire sans aucune garantie pour les paquets d'arriver à bon port et dans un délai raisonnable : les paquets peuvent être perdus, détruits en cas de congestion, de bouclage, ou d'erreur de transmission.

Nous rappelons rapidement la terminologie utilisée dans l'environnement Internet : le protocole IP a pour but de transmettre les paquets de données d'un hôte émetteur à un hôte récepteur, à travers un réseau composé d'éléments intermédiaires et de sous-réseaux. Les éléments intermédiaires sont appelés des **routeurs** et ont la capacité de déterminer l'interface sur laquelle relayer un paquet en fonction de sa destination (fonction de calcul de route ou de **routage**), puis de relayer celui-ci sur cette interface (fonction de **transmission**). Tous les éléments sont identifiés de manière unique par les adresses IP associées à leurs interfaces. Ces adresses contiennent aussi une



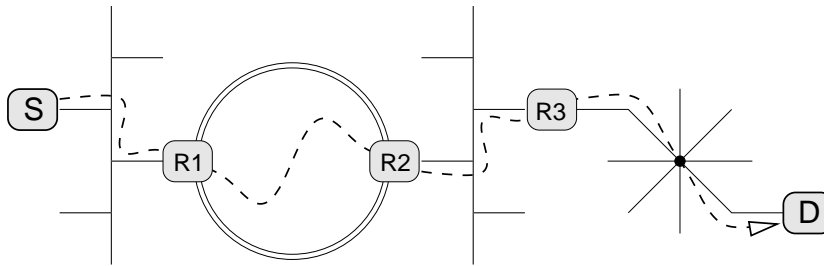


Fig. 2.1 L'interconnexion de sous-réseaux.

information topologique qui permet le routage des paquets vers une destination (transmission “1-vers-1”).

La Figure 2.1 présente le trajet d'un paquet d'un hôte source S vers un hôte destination D à travers plusieurs routeurs. Au niveau d'un sous-réseau, la correspondance des adresses IP avec celles de la couche inférieure se fait avec le protocole ARP (*Address Resolution Protocol*). Cette résolution dépend du médium utilisé et permet, une fois les adresses résolues, la transmission à l'intérieur du sous-réseau. Dans l'exemple, la résolution d'adresses pour transmettre au routeur R1 est effectuée pour permettre l'envoi de données vers celui-ci. De la même manière, la résolution recommence dans chaque sous-réseau jusqu'à la destination D. Les paquets sont transmis de routeur en routeur tant que l'on n'atteint pas le sous-réseau où se trouve le destinataire.

Un autre principe de base de l'Internet est le **principe du bout en bout** qui consiste à reporter les décisions finales chez l'utilisateur et à minimiser l'intelligence à introduire dans le réseau pour éviter toute fonction de contrôle redondante [SRC84]. De ce fait, la plupart des technologies de réseau, sur lesquelles l'architecture Internet repose, ne fournissent aucune garantie de service, que ce soit du point de vue des délais que de celui de la fiabilité. La qualité de service sera généralement assurée par les couches protocolaires supérieures, au sein de l'hôte.

Le service de base fourni dans l'Internet est donc **“Au Mieux”**. Initialement, différents services devaient être fournis : le deuxième octet de l'en-tête IP, appelé champ **Type Of Service (TOS)** est composé de deux sous-champs relatifs aux priorités et au type de service proprement dit. Le premier permet de spécifier huit niveaux de priorités influant sur la gestion des files d'attente, et le second permet de spécifier un type de chemin (avec le débit le plus élevé, le délai le plus faible, la meilleure fiabilité ou le coût minimum) et influe sur le routage, en cas d'utilisation d'un protocole de routage avec plusieurs métriques tels que OSPF ou BGP. L'introduction du champ TOS ne modifie pas la nature du service Internet qui reste *Best Effort* mais permet de l'adapter selon les besoins des utilisateurs. L'utilisation de priorités est discutable et de manière générale le champ TOS n'est pas utilisé dans l'Internet actuel, à l'exception récente de propositions visant l'introduction de service différencié au sein du réseau.

### 2.2.2 Services IP avec Multicast

Les applications actuelles de l'Internet reposent sur un modèle de transmissions de type **un émetteur vers un récepteur** ; et pour cause, IP ne gère que ce type de communication dans sa version de base. Pourtant de nombreuses applications ont besoin d'un modèle de communication plus élaboré, appelé **multipoint** et visant les transmissions d'un hôte vers plusieurs (1-vers-N), voir de plusieurs vers plusieurs (N-vers-M).

L'absence de fonctionnalités multipoints a entraîné l'émulation de ce service pour certaines applications de bases de l'Internet, telles que la messagerie électronique (listes de diffusion), les forums de discussion (news), les discussions en ligne (IRC) ou la distribution de logiciels avec serveurs intermédiaires (sites miroirs). Pour pallier le manque de fonctionnalité multipoint, divers mécanismes ont été utilisés, tels que la mise en place séquentielle ou parallèle de transmissions 1-vers-1 et la gestion de serveurs de réplication intermédiaires.

Au sein de l'Internet, les communications multipoints sont justifiées par trois propriétés importantes :

**Diffusion sélective** La réponse aux besoins multipoints des applications existantes et utilisant le mode point-à-point d'une part et la satisfaction des demandes des nouvelles applications basées directement sur le mode multipoint d'autre part.

**Découverte de ressources** A l'aide de requêtes destinées à un membre d'un groupe, on peut découvrir implicitement le destinataire en diffusant une demande de réponse vers tous les membres du groupe<sup>1</sup> ;

**Efficacité** Les communications multipoints engendrent une réduction de la charge du réseau grâce à la suppression des duplications de paquets dans les tronçons communs (l'émulation du multipoint est particulièrement inefficace) ;

L'intégration du multipoint au sein de l'IP actuel est le résultat des travaux de recherche menés par Deering à l'Université de Stanford [Dee91]. Le multipoint est présenté sous la forme d'une extension appelée *IP Multicast*. Dans la suite, nous parlons de **multicast** pour nommer les fonctionnalités multipoints associées à IP. Le modèle IP de base, avec ses transmissions vers un destinataire unique, sert un modèle de communication appelé **unicast** par opposition aux transmissions vers de multiples destinataires du **multicast**.

L'extension multicast a été introduite avec la définition des **adresses de groupes** [DC84]. Les adresses de groupes permettent un découplage entre les émetteurs et les récepteurs dans une communication *multicast*. Les émetteurs envoient leurs paquets vers une adresse virtuelle, de la même manière que vers une adresse IP classique, sans avoir la nécessité de connaître le groupe de destinataires atteint à travers cette

---

<sup>1</sup>Pour connaître le routeur d'un sous-réseau, une interface peut diffuser sa demande vers un groupe correspondant aux routeurs et auquel le routeur en question se sera abonné.

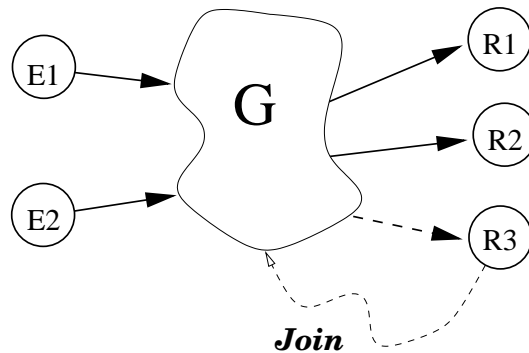


Fig. 2.2 L'abstraction de groupe avec IP multicast.

adresse. Les récepteurs acceptent volontairement les paquets destinés à cette adresse en adhérant au groupe correspondant à celle-ci.

La Figure 2.2 synthétise le concept de groupe : des émetteurs, E1 et E2 émettent à destination du groupe G, et les récepteurs participant au groupe à cet instant (R1, R2 et R3) reçoivent les paquets destinés à celui-ci. La notion d'appartenance est dynamique, la liste des membres du groupe étant variable au cours du temps. Un membre (ici R3) peut à tout moment quitter ou rejoindre le groupe à l'aide d'une action appelée *Join*. La gestion de l'appartenance à un groupe se fait de manière autonome et indépendante et ne doit pas influencer sur les autres membres du groupe.

La notion d'abstraction de groupe ainsi définie, propose un modèle de communication N-vers-M, aussi appelé *many-to-many* dans le cadre du mode datagramme, car plusieurs émetteurs peuvent émettre simultanément vers l'adresse de groupe, elle-même destinée à atteindre plusieurs destinataires<sup>2</sup>.

La mise en place des fonctionnalités IP Multicast repose sur une signalisation : IGMP (*Internet Group Management Protocol*) [Dee89] et la définition des adresses de groupes. Sous IP, une partie de l'espace d'adressage est dévolue à ces adresses multicast, appelées **Adresses de Classe D**. Ces adresses permettent l'abstraction entre le contenu et le contenant, et ainsi d'adresser virtuellement toutes les interfaces Internet qui auront décidé de se joindre au groupe correspondant. Le service IP Multicast est dit **orienté récepteur** car l'adhésion à un groupe d'un récepteur ne dépend que de lui-même et ne fait pas intervenir les émetteurs (qui ignorent la composition de ce groupe).

Un émetteur envoie des paquets vers un groupe multicast sans avoir d'informations sur ses participants, et un récepteur appartient à un groupe sans explicitement connaître les émetteurs du groupe. La gestion du groupe par le réseau et non par

<sup>2</sup>La réception à partir de plusieurs émetteurs est aussi une caractéristique des transmissions *unicast*, que l'on considère généralement basées sur le modèle 1-vers-1 (*one-to-one*). Mais, de part le mode non connecté et la possibilité que plusieurs émetteurs atteignent la même adresse, ces transmissions peuvent être associés à un modèle N-vers-1 (*many-to-one*).

l'émetteur permet une bonne adaptation au facteur d'échelle. L'attachement d'un membre et son départ, sont des primitives de IGMP appelées par le membre lui-même. Le processus est local, c'est le protocole de routage multicast qui gère la distribution des paquets selon les groupes. Le système est basé sur des envois périodiques de messages afin d'assurer la dynamique du système.

La structure de l'Internet pour laquelle IP a été définie repose sur des réseaux à média partagés, tels *Ethernet*, et des liaisons point-à-point. Le multicast IP a été initialement dimensionné pour ces réseaux possédant des capacités de diffusion natives. Pour des réseaux ne possédant pas ces capacités, tels ceux de type maillé, la gestion du multicast est plus complexe. Nous étudierons dans la section 2.3 le cas des réseaux ATM.

La validation la plus visible et la plus étendue de l'extension multicast a été effectuée au sein d'un réseau multicast expérimental appelé **MBone**. Celui-ci est opérationnel depuis début 1992 et est principalement utilisé pour la diffusion multimédia (audio et vidéo) et le travail coopératif (tableau blanc distribué).

La structure du MBone repose sur un réseau virtuel superposé à celui de l'Internet. Elle est composée de zones ayant des capacités multicast interconnectées par des tunnels pour pouvoir traverser le reste de l'Internet où le multicast n'est pas disponible. Typiquement, les zones multicast sont composées de LAN intégrant le multicast de manière native (tels que les réseaux locaux de type Ethernet) et reliés entre eux par des routeurs intégrant IP Multicast ou **mroute**s. Les tunnels se situent entre deux mrouteurs de zones multicast différentes et procèdent à l'encapsulation des paquets IP multicast dans des paquets IP classiques avec une adresse unicast.

Comme références sur le MBone, on peut citer l'article de Casner qui relate l'une des toutes premières expériences [Cas92]. Un article d'Eriksson [Eri94] permet une vision plus globale des possibilités, et le livre de Kumar [Kum96] permet d'approfondir le domaine.

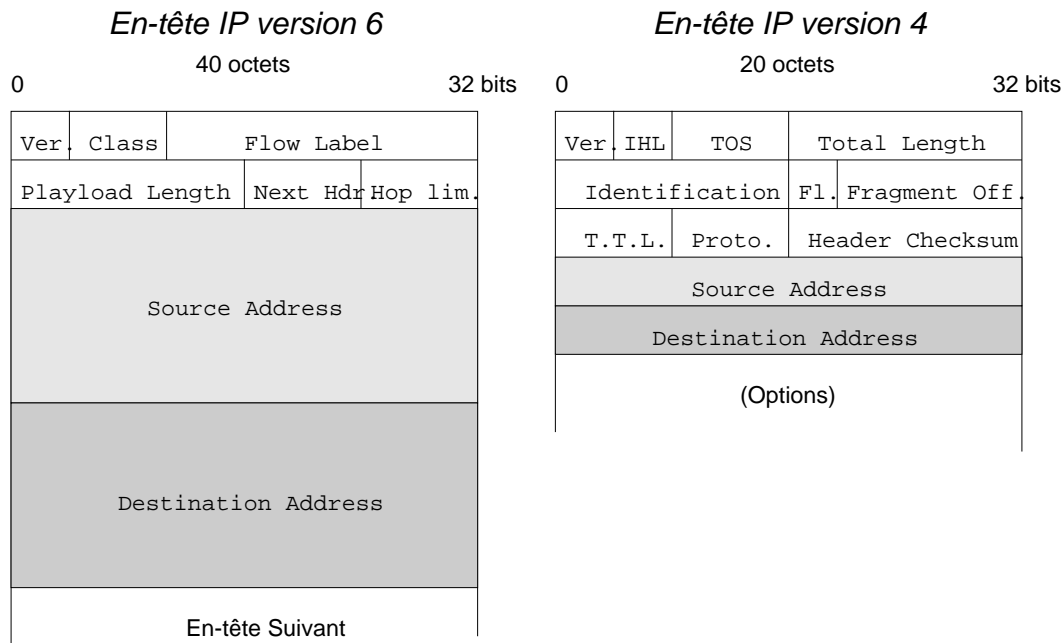
La qualité de service fournie par IP Multicast est identique à celle fournie par IP classique : il n'y en a pas, le service est *Best Effort*.

### 2.2.3 Services IP version 6

Aujourd'hui, l'Internet repose sur IP version 4 (IPv4). Ce protocole a été défini à une époque où l'on ne pouvait prévoir son succès actuel. Deux problèmes cruciaux apparaissent :

- L'épuisement des adresses (pour certaines classes de sous-réseaux) ;
- L'explosion de la taille des tables de routage.

La solution proposée a été l'introduction d'une nouvelle version du protocole : IP version 6 (**IPv6**) [DH95]. Celle-ci est une évolution d'IPv4 dont elle reprend les



**Fig. 2.3** Comparaison des en-têtes de paquets IP.

principaux mécanismes avec l'introduction de concepts récents pour un traitement plus efficace et une pérennité accrue :

**Adresses sur 128 bits** Quadruplement de la taille des adresses avec redéfinition globale de la structure de l'adressage et intégration des concepts d'adressage agrégé de CIDR (*Classless Internet Domain Routing*) [FLYV93] ;

**Taille fixe de l'en-tête** avec gestion des options par encapsulation (Chaînage d'en-tête) Permet un traitement rapide de la plupart des paquets ;

**Pas de Checksum** L'abandon du contrôle d'erreur de l'en-tête IP permet un relai plus performant au niveau des routeurs qui n'ont plus à le calculer ;

**Pas de segmentation** Découverte de la taille maximum des paquets (MTU pour *Maximum Transmission Unit*) pour un chemin ;

**Class et Flow Label** Introduction de nouveaux champs pour aider la gestion de différents services (leur utilisation n'est pas encore fixée).

La Figure 2.3 présente les différences principales entre les en-têtes des deux versions.

Les changements de taille de certains champs, notamment ceux des adresses, nécessitent la modification des API (*Application Programming Interface*) [GaJBS97] et des protocoles associés à IPv4. La résolution d'adresses fait partie de la base du protocole IPv6 et est donc située au niveau réseau plutôt que dans les niveaux inférieurs. Les mécanismes d'adaptation d'IPv4 sur ATM ne pourront être traduits directement pour IPv6 à cause de ce transfert de fonctionnalités. Le nouveau protocole ND (*Neighbor Discovery*) [NNS96], qui fait partie intégrante d'IPv6, remplit les

fonctionnalités de résolution d'adresses assurées précédemment par ARP. Le Multicast est intégré d'origine avec IPv6 et sa gestion a été intégrée au protocole de contrôle : IGMP fait partie de ICMPv6 (*Internet Control Message Protocol* pour IPv6) [CD95].

Au niveau expérimental, les validations de la nouvelle version, de tous ses protocoles associés et des mécanismes de migration sont effectuées au sein d'un réseau expérimental, le **6Bone**, construit sur le même principe que le Mbone.

Le service de base prévu par IPv6 reste "Au Mieux", mais deux champs ont été intégrés pour offrir des services différents pour certains flots de données. Le champ **Class** est en cours de spécification. Il devrait permettre l'introduction de différenciation de service au sein du réseau. Le champ **Flow Label** doit aider à l'identification des flots pour les trafics qui nécessitent un traitement particulier tels ceux associés à RSVP, par exemple, le protocole présenté dans la prochaine section. Ce champ permet uniquement une meilleure gestion du trafic dans les routeurs et n'introduit pas la notion d'état dans le réseau IPv6.

Pour une description plus détaillée d'IPv6 se reporter au livre de Huitéma [Hui98]. Pour une présentation des implémentations d'IPv6 voir [Ciz98].

#### 2.2.4 Services IP avec QoS

Le protocole IP ne gère pas d'état du réseau dans les nœuds intermédiaires. Tout paquet est acheminé vers son destinataire de manière indépendante. Il est impossible de garantir à un flot de données particulier un délai d'acheminement borné. Il peut donc paraître nécessaire d'ajouter à IP un protocole de signalisation qui pourra gérer un minimum d'état dans le réseau, tel que RSVP (*Resource Reservation Protocol*) [ZDE<sup>+</sup>93]. Celui-ci est utilisé pour indiquer la qualité de service à fournir pour un flot de données. Le service fourni initialement était identique vers tous les récepteurs : le service *Best Effort*. Le but du protocole RSVP est d'assurer la signalisation pour fournir un service multipoint-à-multipoint avec une QoS adaptée (hétérogène) à chaque destinataire et renégociable de manière dynamique vers chacun d'entre eux.

Le groupe de recherche RSVP de l'IETF a spécifié le protocole dans [BZB<sup>+</sup>97, Wro97b]. RSVP assure la signalisation de réservations de ressources sur IP multicast mais pas la mise en œuvre de celles-ci. RSVP est utilisé pour indiquer la qualité de service à fournir pour un flot de données. Les paramètres de QoS sont spécifiés par un autre groupe de recherche de l'IETF, appelé **IntServ** (*Internet Services*), qui a défini deux classes de services [BCS94, Wro97a, SPG97] dont le Tableau 2.1 fait la description rapide.

#### Fonctionnalités de RSVP

Les caractéristiques du protocole RSVP sont réunies dans le Tableau 2.2.

Classe	Description
“Guaranteed Services”	La classe à <i>délat garanti</i> fournit des bornes strictes sur le délai maximum d’acheminement des paquets de bout en bout.
“Controlled Load”	La classe à <i>charge contrôlée</i> procure le service minimal : identique à celui qui serait procuré en utilisant le service “au mieux” habituel sur le réseau non congestionné.

**Tab. 2.1** Les classes de service de l’IntServ

Fonctionnalité	Description
Signalisation IP	RSVP fonctionne sur IP au niveau transport en tant que signalisation intégrée avec les fonctionnalités de routage, de contrôle d’admission, de filtrage et d’ordonnancement.
Orienté récepteur	Les récepteurs d’un flot gèrent les réservations associées à celui-ci.
Unidirectionnel	Réservation pour des flots unidirectionnels.
N-vers-M	Gestion des réservations jusqu’au modèle multipoint-à-multipoint.
Dynamacité	Adaptation dynamique aux changements de routages et de groupes.
Soft State	Gestion d’un état relâché dans le réseau mis à jour périodiquement.
Agrégation	Différents styles de réservation pour la fusion des flots.
Transparence	Prise en compte des routeurs non RSVP.

**Tab. 2.2** Les fonctionnalités de RSVP

## Mécanismes de RSVP

Les réservations sont basées sur des **sessions** identifiant un flot de données pour une destination et un protocole de transport. RSVP utilise principalement deux messages : **PATH** et **RESV**.

- Les messages PATH sont envoyés des émetteurs vers les récepteurs, établissent un état dans les nœuds traversés et transmettent la description des flots des émetteurs aux récepteurs. Les messages PATH suivent les mêmes chemins que les données et construisent un arbre de parcours inverse que suivront les messages RESV.
- Les messages RESV sont renvoyés des récepteurs vers les émetteurs par les chemins de retour définis avec les messages PATH. Les réservations sont réalisées à l'aide des messages RESV en une passe, avec possibilité d'obtenir des informations préalables à travers le champ **Adspec** des messages PATH (cette technique est appelée OPWA pour *One Path With Advertising*).

Les messages sont émis périodiquement afin de maintenir dynamiquement les chemins et les réservations utilisées. Cette technique permet à RSVP de s'adapter à toute modification liée au transport des paquets. L'état relâché maintenu dans le réseau à l'aide d'envoi de paquets périodique est appelé *Soft State*, par opposition à l'état "en dur" utilisé par les protocoles basés sur la commutation de circuit. Deux messages, PTEAR et RTEAR, permettent de libérer les réservations explicitement. L'expiration des temporisateurs associés aux réservations permettent la libération implicite de celles-ci.

Une réservation RSVP est associée à un descripteur de flot, composé d'un **FlowSpec** et d'un **FilterSpec** :

**FlowSpec** Objet contenant une description de la classe de service et deux valeurs numériques : **Rspec** et **Tspec** représentant respectivement la qualité de service et le trafic (*token bucket*).

**FilterSpec** Objet utilisé pour choisir un sous-ensemble de paquets du flot<sup>3</sup>.

Un **style** est associé à une réservation, ce qui permet d'effectuer des agrégations entre les différents flots d'une session. Les paramètres sont fonction de la capacité des flots à partager une même ressource par émission alternative ou non (*shared* ou *distinct*) et de l'expression explicite des émetteurs ou non (*explicit* ou *wildcard*). Trois styles sont définis, nous les décrivons dans le Tableau 2.3.

## Architecture RSVP

Une architecture basée sur RSVP pour fournir des services intégrés repose sur différents éléments représentés sur la Figure 2.4 : module de routage, module RSVP, module de classification, ordonnanceur de paquets et contrôleur d'admission.

---

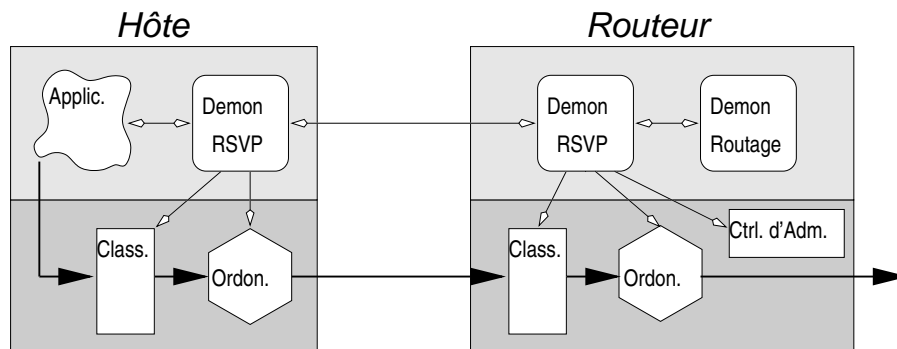
<sup>3</sup>La sélection est effectuée par rapport à l'émetteur ou par le champ d'un protocole de niveau supérieur.



Style	Représentation	
<b>Wildcard-Filter</b>	$WF(*Q)$	Une réservation est partagée par un groupe d'émetteurs (la réservation correspond à la demande la plus importante).
<b>Shared-Explicit</b>	$SE((S_1, S_2, \dots)Q)$	Idem que précédemment mais les émetteurs sont déclarés explicitement.
<b>Fixed-Filter</b>	$FF(S_1 Q_1), FF(S_2 Q_2), \dots$	Une réservation distincte par émetteur (pas de partage, la réservation correspond à la somme des demandes).

\* pour l'émetteur quelconque,  $Q$  pour le flowspec demandé,  $S_n$  les émetteurs nommés et  $Q_n$  pour le flowspec correspondant.

**Tab. 2.3** Les styles RSVP



**Fig. 2.4** Architecture des nœuds du réseau avec RSVP.

Lors de l'arrivée d'un message RESV pour une session, le module de contrôle d'admission accepte ou non la requête. En cas de succès, le `FilterSpec` est intégré dans le module de classification pour qu'il puisse identifier les paquets appartenant à la session, et le `FlowSpec` est transmis à l'ordonnanceur pour qu'il puisse assurer la QoS relative aux paquets de cette session. Lorsqu'un paquet utilisant la réservation arrive ensuite, il est associé aux ressources de la session par le classificateur et est transmis à l'ordonnanceur de paquets.

## 2.3 LES SERVICES DES RÉSEAUX ATM

Les fondements de la technologie ATM se situent au début des années 80. Après une quinzaine d'années de développement, les réseaux basés sur cette technologie que l'on avait annoncés comme étant la solution pour l'intégration de service, reviennent à une situation plus réaliste et représentent une solution complémentaire.

ATM est la technologie adoptée par l'ITU-T pour son Réseau Numérique à Intégration de Services à Large Bande (RNIS-LB ou *B-ISDN* en anglais). Elle a été choisie pour ses capacités à fournir tous les types de services, de la transmission audio ou vidéo aux transferts de données informatiques, le tout sur un médium unique. Cette technologie originaire du monde des télécommunications répond à un besoin d'intégration stimulé par les opérateurs qui veulent rompre avec la gestion de réseaux différents pour chaque service (téléphonie, vidéo, données...).

Notre intérêt pour la technologie ATM repose principalement sur les caractéristiques originales en matière de service et les potentialités futures en terme d'adaptation aux nouveaux besoins, que ce soit en terme de bande passante ou de nouvelles fonctionnalités.

Nous présentons dans la suite les principes de base et détaillons les services fournis, particulièrement pour le multipoint. Pour approfondir, citons un article [CPS94] de Coudreuse, l'un des promoteurs de l'ATM, et le livre de De Prycker [dP95].

### 2.3.1 Principes de base d'ATM

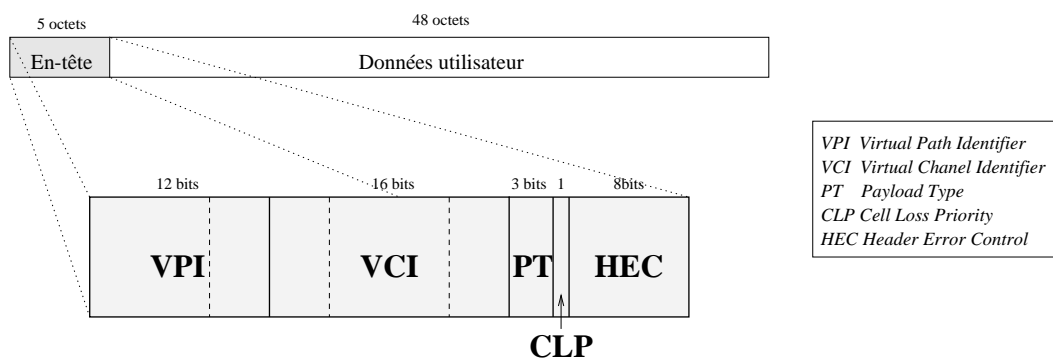
La technologie ATM est, comme son nom l'indique, basée sur le "mode de transfert asynchrone", par opposition à la plupart des autres technologies de réseaux d'opérateur qui sont basées sur le mode synchrone appelé STM (*Synchronous transfer mode*) et basées sur une horloge réseau globale.

L'utilisation du mode asynchrone implique un multiplexage par étiquette, se rapprochant du mode datagramme, plutôt qu'un multiplexage temporel. L'information est ainsi véhiculée dans de petits paquets de données étiquetés et de même taille appelés **cellules**. La taille fixe et réduite des cellules permet la commutation effi-

cace de celles-ci, ouvrant ainsi l'accès aux très hauts débits et aux temps de latence réduits.

## La cellule ATM

L'unité de transmission ATM est la cellule. La taille de celle-ci est de 53 octets, dont 48 pour les données utilisateur et 5 pour l'en-tête qui contient l'étiquette citée précédemment.



**Fig. 2.5** Structure d'une cellule ATM

Dans la Figure 2.5 est représentée une cellule ATM dont on a extrait l'en-tête pour énumérer les champs qui le composent :

**VPI** Le champ *Virtual Path Identifier* code sur 12 bits l'identificateur de conduits virtuels (identification de 4.096 conduits au maximum).

**VCI** Le champ *Virtual Channel Identifier* code sur 16 bits l'identificateur de voies virtuelles (identification de 65.535 voies au maximum dans chaque conduit).

**PT** Les 3 bits du champ *Payload Type* codent le type d'information transportée par la cellule. Cette information permet de distinguer les cellules de contrôle de celles de données.

**CLP** Le bit du champ *Cell Loss Priority* indique si la cellule est importante. En cas de congestion, les cellules de moindre importance seront éliminées.

**HEC** Le dernier octet est une zone de contrôle appelée *Header Error Control* et destinée à protéger l'en-tête.

Le couple VPI/VCI identifie de manière unique les cellules au niveau d'un lien. Les communications vont être effectuées en identifiant les cellules d'une transmission pour chaque lien qu'elle traverse et en utilisant une table de translation dans chaque élément intermédiaire reliant ces liens. Cette suite de couples VPI/VCI va former une connexion virtuelle.

## Les connexions virtuelles ATM

La connexion virtuelle ATM est la transposition de la notion de circuit du monde des télécommunications dans le contexte de la commutation de cellule. La commutation ATM utilise un mode non connecté qui ne possède qu'un adressage local (adresse au niveau du lien), il est donc nécessaire d'introduire une procédure qui va initier les tables de translation de VPI/VCI d'un lien à l'autre pour un flot donné à l'aide des adresses ATM globales. La procédure d'initialisation va créer un état au sein des éléments intermédiaires du réseau, pour toute la durée de la connexion et présenter ainsi ATM comme une technologie **orientée connexion**.

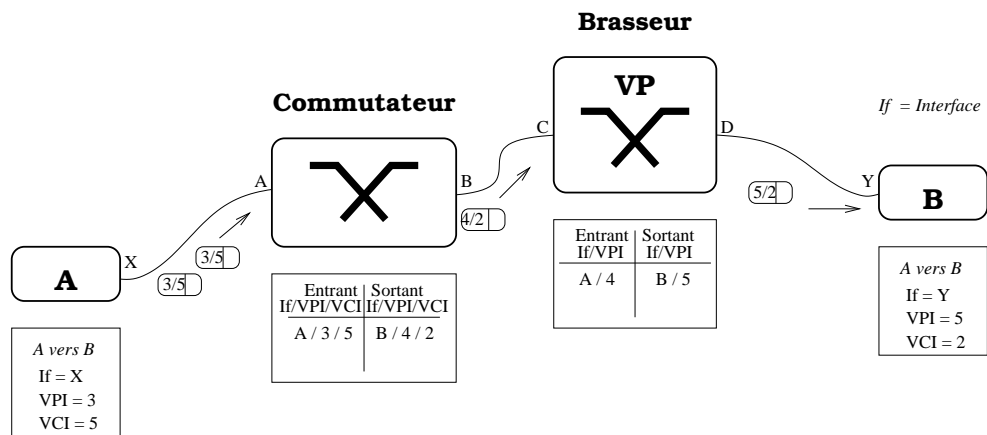


Fig. 2.6 Principe de la commutation de cellules

La Figure 2.6 présente une connexion ATM à travers les deux types d'éléments de commutation d'un réseau ATM :

**Commutateur ATM** Élément du réseau effectuant la correspondance entre les triplets port/VPI/VCI d'entrée et de sortie. Le commutateur travaille au niveau des voies virtuelles.

**Brasseur ATM** Élément du réseau effectuant la correspondance entre les couples port/VPI d'entrée et de sortie. Le brasseur travaille au niveau des conduits virtuels.

Les interfaces au niveau d'un élément ATM sont appelées des ports. L'introduction des conduits et des voies virtuelles forme une hiérarchie de commutation et permet, par exemple, de séparer un niveau de conduits permanents gérés par un opérateur, d'un niveau de voies gérées par l'utilisateur.

## La pile protocolaire ATM

Le modèle de référence des réseaux ATM est représenté dans la Figure 2.7. Il comporte 3 plans découpés en plusieurs couches, comme le stipule la recommandation

I.321 [CCI91]. Les deux plans arrières sont des plans des gestions associés à l'administration des différentes couches et des plans eux-mêmes. Le premier plan est subdivisé en deux autres plans, le **plan contrôle** pour le trafic relatif à la signalisation et le **plan utilisateur** pour le trafic de données.

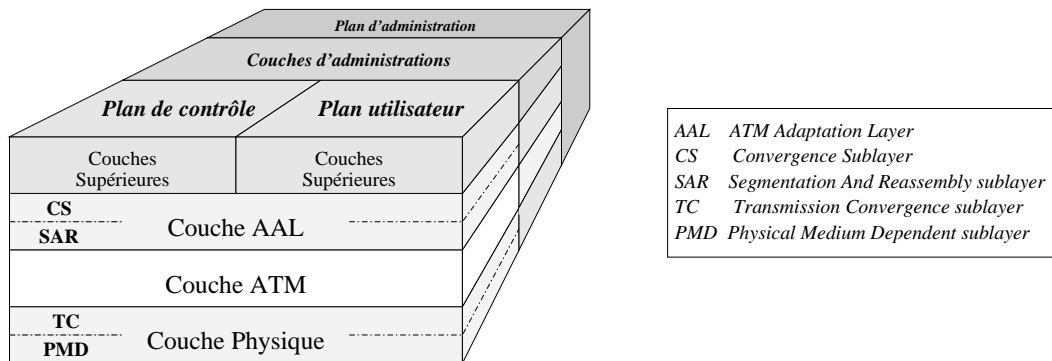


Fig. 2.7 La pile protocolaire ATM

Les couches qui composent ces plans sont, dans l'ordre, en partant du média :

**Couche Physique** couche ayant la charge de l'accès au support physique (fibre optique par exemple). Elle se compose de deux sous couches :

**PMD** sous-couche spécifique au médium concerné.

**TC** sous-couche de convergence gérant les fonctionnalités communes aux couches physiques ayant généralement rapport avec la conversion des cellules en bits.

**Couche ATM** couche totalement indépendante du médium, assurant les fonctions de commutation et multiplexage des cellules grâce aux VPI et VCI ainsi que le contrôle de flux. La couche ATM constitue le traitement principal des éléments de commutation d'un réseau ATM.

**Couche AAL** couche que l'on retrouve aux extrémités du réseau pour fournir un service adapté aux niveaux supérieurs. Elle se compose de deux sous couches :

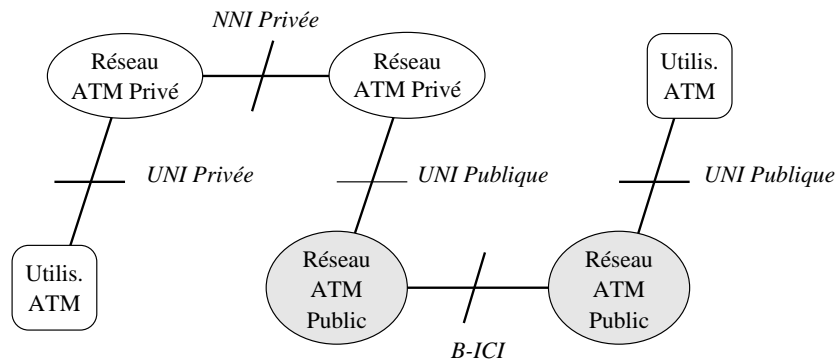
**SAR** sous-couche de segmentation des AAL PDU (*Protocol Data Unit*) en cellules et de réassemblage.

**CS** sous-couche de convergence.

**Couches supérieures** couches s'appuyant sur le service AAL.

### 2.3.2 Services offerts à travers UNI

La normalisation des réseaux ATM se base sur différentes interfaces pour séparer les entités du réseau et spécifier les règles d'interfonctionnement. Les principales interfaces définies sont représentées dans la Figure 2.8.

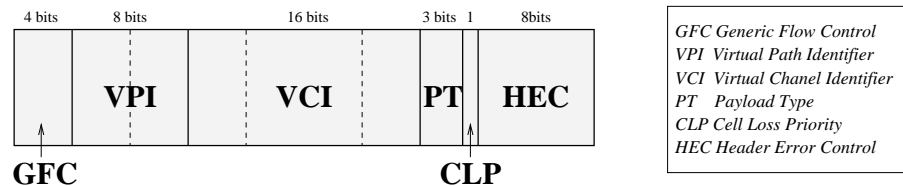


**Fig. 2.8** Les différentes interfaces des réseaux ATM

Deux catégories d'interfaces sont présentes, les interfaces entre l'utilisateur et le réseau, appelées UNI (*User-Network Interface*) et les interfaces entre les réseaux. Si elles sont privées, on parle de PNNI (*Private Network-Network Interface*), et si elles sont publiques de B-ICI (*B-ISDN InterCarrier Interface*). On trouve généralement l'interface UNI au niveau de la liaison d'un adaptateur ATM d'un hôte avec un commutateur et PNNI entre des commutateurs.

L'interface qui nous intéresse est l'UNI, car nous étudions les services fournis au niveau d'un hôte. Deux types d'UNI existent, selon que l'on s'interface à un réseau public ou privé. L'UNI publique se retrouve dans deux cas spécifiques : au niveau d'un hôte directement raccordé au réseau public, ou entre un réseau public et un réseau privé<sup>4</sup>. L'UNI privée se rencontre dans tous les autres cas. La différence entre les UNI privées ou publiques repose sur quelques fonctionnalités optionnelles dans les unes et obligatoires dans les autres ; les spécifications de base restant les mêmes.

Les mécanismes offerts par les deux UNI au niveau ATM correspondent à ceux présentés précédemment, à ceci près que l'en-tête de la cellule ATM est légèrement modifié, avec un nouveau champ. La Figure 2.9 situe ce champ de 4 bits appelé GFC (*Generic Flow Control*). La réduction à 8 bits du champs VPI limite à 256 le nombre de conduits virtuels, mais est censée permettre la mise en œuvre d'un mécanisme de contrôle de flux des cellules des applications au niveau local.



**Fig. 2.9** Structure d'une cellules ATM, à travers UNI

<sup>4</sup>L'accès à un réseau public se fait systématiquement à travers l'interface UNI publique

L'interface UNI a pour but principal de permettre l'utilisation de connexions permanentes ou non au sein du réseau ATM. Une connexion permanente sera appelée *Permanent Virtual Connection* (PVC) et sera gérée de manière spécifique à chaque commutateur. Une connexion dynamique (ou commutée) sera appelée *Switched Virtual Connection* (SVC) et sera établie à partir de caractéristiques spécifiées à travers l'UNI à l'établissement. Le réseau peut accepter ou refuser un SVC. S'il l'accepte, il fournit à l'utilisateur une valeur de VPI/VCI à utiliser pour tous les transferts relatifs à cette connexion jusqu'à la terminaison et la libération des ressources.

Les signalisations UNI que nous présentons dans la suite sont celles définies au sein de l'**ATM Forum**. En particulier, nous allons présenter UNI 3.0 et UNI 3.1 que l'on trouve actuellement au sein des éléments ATM, puis UNI 4.0 qui est la version la plus récente.

### Interfaces UNI 3.0 et 3.1

Les premières versions de l'UNI ont été publiées en 1993 pour la version 3.0 [UNI93] et 1994 pour son évolution, la version 3.1 [UNI94]. La version 3.1 corrige la version 3.0 tout en se conformant à la signalisation définie dans la recommandation Q.2931 [ITU95] par l'ITU-T. Dans ces versions de l'UNI, les couches protocolaires adressées vont du niveau physique aux couches supérieures en passant par la couche ATM, les plans de contrôle et la gestion de trafic.

L'interface UNI 3.x permet d'initier, de gérer et de terminer des connexions virtuelles. Celles-ci ont pour caractéristique principale d'être soit point-à-point et bidirectionnelles, soit point-à-multipoint et unidirectionnelles. La Figure 2.10 représente ces deux types de connexions à travers un réseau ATM pour lequel nous avons fait abstraction de la topologie, et que l'on appellera dans la suite **nuage ATM**. Nous reviendrons sur le service multipoint dans la section 2.3.3.

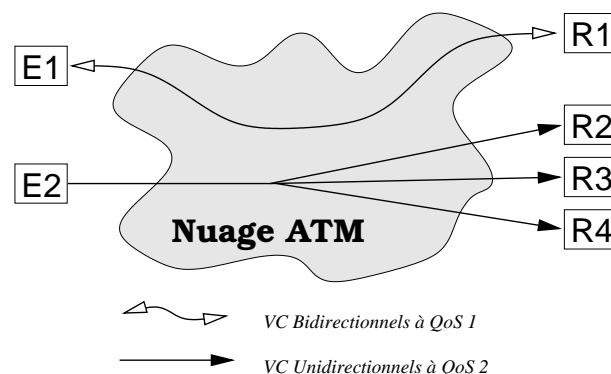


Fig. 2.10 Services de base

Les connexions virtuelles sont définies au niveau conduit virtuel (VP) ou/et au niveau voie virtuelle (VC). Les connexions peuvent posséder une classe de service parmi les deux suivantes :

**CBR** (*Constant Bit Rate* aussi appelé BCOB-A) Classe de service où les paramètres de QoS spécifiés sont le débit crête d'émission des cellules et la variation de délai. Cette classe est adaptée aux flux temps réel à débit constant.

**VBR** (*Variable Bit Rate* aussi appelé BCOB-C) Classe de service où les paramètres de QoS spécifiés sont le débit moyen, le débit crête d'émission des cellules et la taille des rafales de cellules. Cette classe n'est pas adaptée aux flux temps réel et vise des débits variables.

On assimile l'absence de classe de service à une nouvelle classe :

**UBR** (*Unspecified Bit Rate*) Classe de service sans paramètres de QoS spécifié, que l'on peut comparer à un service "Au Mieux". Cette classe est utilisée pour les flux sans garantie de service tels que les flux de données informatiques.

Les paramètres de QoS sont fixés à la création d'une connexion et ne sont pas renégociables.

#### Interface UNI 4.0

La version actuelle de l'UNI a été publiée en 1996, et contrairement aux versions monolithiques précédentes, est divisée en plusieurs documents : un pour chaque interface physique, un pour la signalisation UNI 4.0 elle-même (SIG 4.0) [UNI96], un pour la gestion de trafic, appelé *Traffic Management* (TM 4.0) [TM96] et un pour l'administration de l'interface, appelé *Integrated Local Management Interface* (ILMI 4.0) [ILM96]. Le service fourni par l'UNI 4.0 est une évolution des services fournis précédemment avec de nouvelles fonctionnalités, telles que la fonctionnalité d'accès orienté récepteur à un arbre de diffusion sur lequel nous reviendrons dans la section 2.3.3. De nouvelles classes de services sont définies :

**CBR** (*Constant Bit Rate*) Classe définie précédemment.

**rt-VBR** (*Real-Time Variable Bit Rate*) Nouvelle classe VBR avec contrainte sur la variation du délai de transmission.

**nrt-VBR** (*Non Real-Time Variable Bit Rate*) Classe définie précédemment sous le nom de VBR.

**ABR** (*Available Bit Rate*) Nouvelle classe pour les applications pouvant moduler leur débit, basée sur un contrôle de flux réactif. Non spécifiée pour le point-à-multipoint.

**UBR** (*Unspecified Bit Rate*) Classe définie précédemment.



Acronyme	Description
PCR	Le <i>Peak Cell Rate</i> indique le débit d'émission maximum des cellules.
SCR	Le <i>Sustainable Cell Rate</i> indique une borne supérieure du débit d'émission moyen des cellules.
MCR	Le <i>Minimum Cell Rate</i> indique le débit d'émission de cellules minimum.
MBS	Le <i>Maximum Burst Size</i> indique la taille maximum des agrégats de cellules.
CDVT	Le <i>Cell Delay Variation Tolerance</i> indique la gigue maximum d'une cellule.
CLR	Le <i>Cell Loss Ratio</i> indique le taux de cellules perdues ou arrivées trop tard.
CTD	Le <i>Cell Transfer Delay</i> indique le délai de transmission d'une cellule.
CDV	Le <i>Cell Delay Variation</i> indique la variation de délai de transmission d'une cellule.

Tab. 2.4 Les paramètres de QoS dans les réseaux ATM

### Les paramètres de QoS et les catégories de service ATM

Les interfaces UNI que nous avons présentées utilisent divers paramètres de QoS pour spécifier le service demandé par l'utilisateur. La combinaison de ces divers paramètres permet la gestion des différentes catégories de services. Le Tableau 2.4 présente les principaux paramètres de QoS fournis par les réseaux ATM et le Tableau 2.5, les paramètres utilisés pour chaque catégorie.

Paramètres QoS	CBR	rt-VBR	nrt-VBR	ABR	UBR
CLR pour CLP=0	√	√	√	√	×
CLR pour CLP=1	o	o	o	√	×
CTD	√	√	√	×	×
CDV	√	√	×	×	×
SCR et MBS	¬	√	√	¬	¬
PCR et CDVT	√	√	√	√	√
MCR	¬	¬	¬	√	¬
Contrôle de flux	¬	¬	¬	√	¬

Symboles : "√" spécifié ou utilisé, "o" optionnel, "x" non spécifié, "¬" non applicable.

Tab. 2.5 Les catégories de services des réseaux ATM

### 2.3.3 Services ATM et multipoint

La technologie ATM est conçue pour un support bas niveau de la transmission 1-vers-N. La couche ATM permet la duplication d'une cellule sur plusieurs interfaces. Seul le modèle 1-vers-N est disponible car il est facilement intégré au sein des commutateurs. Le modèle étendu N-vers-M pose le problème de l'entrelacement des cellules de deux trames rejoignant un même port de sortie.

Le service multipoint de base des réseaux ATM propose des connexions 1-vers-N ou **point-à-multipoint**, initiées au niveau de l'émetteur, unidirectionnelles (envoi des cellules seulement en direction des récepteurs) avec la même QoS sur chaque branche de l'arbre de diffusion. Cette QoS est non renégociable.

Le mode orienté connexion des réseaux ATM base la gestion du multicast sur un principe séquentiel d'ajout ou de retrait successif de branches à l'arbre de diffusion. La gestion se fait explicitement au niveau de l'émetteur, appelé racine de l'arbre, qui possède la connaissance de tous les récepteurs. Ce modèle pose des problèmes d'adaptation au facteur d'échelle, tant au niveau temporel, en raison de sérialisation des appels, qu'au niveau ressource, à cause de la centralisation du contrôle.

Le multipoint des réseaux ATM n'offre ni la diffusion, ni l'abstraction des adresses de groupes, ni le routage dynamique du monde IP. Le service ATM se limitant à des connexions point-à-multipoint unidirectionnelles initiées à la racine, deux modèles ont été définis pour gérer un service de type *multicast* au niveau ATM :

### Maillage multicast

Le maillage multicast crée des connexions point-à-multipoint de chaque émetteur vers tous les autres membres du groupe. Ce modèle efficace (liaisons directes) et robuste (distribué) en terme de transmission est très lourd au niveau gestion. Les problèmes proviennent de la consommation de ressources, tant au niveau du nombre de VC utilisés, que de la signalisation globale et de la gestion locale. Le coût de l'ajout d'un membre (en pointillé sur la Figure 2.11) mobilise tous les hôtes impliqués, eux-mêmes devant gérer la liste de tous les autres membres ainsi que les connexions vers ceux-ci.

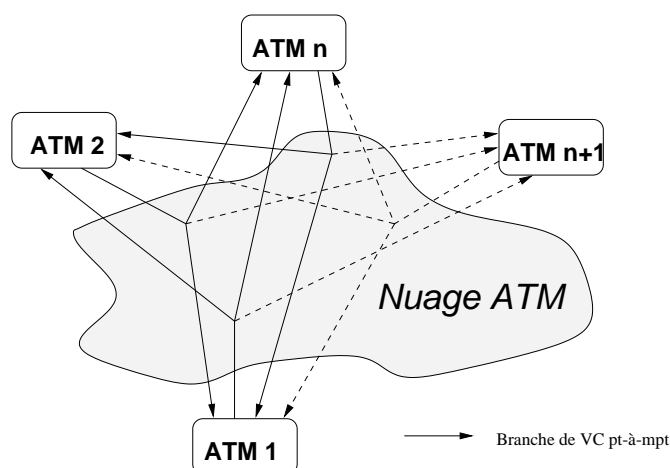


Fig. 2.11 Maillage Multicast ATM

## Serveur multicast

L'approche avec serveur multicast (MCS pour *MultiCast Server*) utilise une entité intermédiaire pour gérer les connexions point-à-multipoint vers lesquelles les émetteurs envoient leur trafic multipoint (principe similaire aux arbres partagés vus dans la section 2.2.2). Cette approche nécessite la création d'une connexion point-à-point de chaque source vers le MCS. Ce dernier établit une connexion point-à-multipoint vers les destinataires correspondants. Le rôle du MCS est de réassembler les paquets arrivant sur les connexions entrantes et de les conserver dans une file d'attente avant de les émettre sur la connexion point-à-multipoint<sup>5</sup>. La gestion de groupes est facilitée par la structure centralisée, par contre cela produit des risques de congestion au niveau du serveur et un routage moins efficace entre les sources et les récepteurs. Ce modèle est mieux adapté que le maillage multicast aux groupes dynamiques de récepteurs (ajout économique d'un membre : en pointillé sur la Figure 2.12) mais le réassemblage des paquets et le double transit vont augmenter le délai d'acheminement.

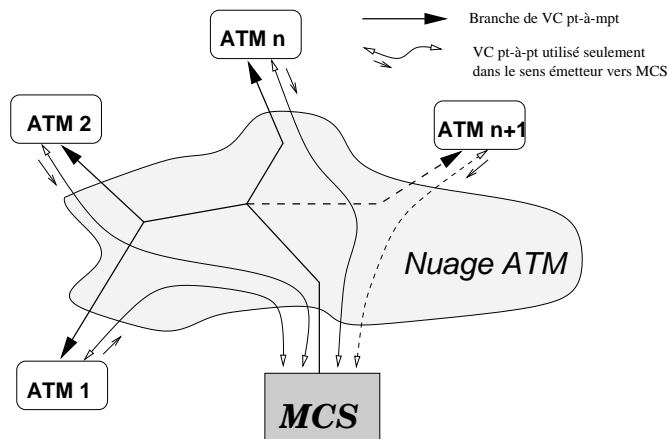


Fig. 2.12 Serveur Multicast ATM

## Extensions du multicast ATM

Dans l'UNI 4.0, une fonctionnalité est apparue pour résoudre en partie les défauts liés à la gestion des arbres multipoint à partir de la racine (RIJ pour *Root Initiated Join*). La nouvelle fonctionnalité permet la création de nouvelles branches à partir des récepteurs (LIJ pour *Leaf Initiated Join*). Mais ce service multipoint est toujours lié à un arbre spécifié par sa racine et ne permet donc pas de découpler émetteurs et récepteurs pour fournir l'abstraction de groupe. De plus, cette fonctionnalité est

<sup>5</sup>La réception de plusieurs paquets simultanément pose problème au niveau du MCS. En effet, si l'on utilise l'AAL5, aucun mécanisme d'ordonnancement n'est disponible et les paquets simultanés vont introduire des délais supplémentaires.

optionnelle. Nous considérons donc que le service multipoint globalement disponible au sein des réseaux ATM est celui décrit précédemment.

Un meilleur support du service *many-to-many* dans ATM est envisagé dans plusieurs propositions :

**Extension du modèle ATM Forum** Au sein de l'ATM Forum, particulièrement dans les groupes de travail relatifs à PNNI, des propositions ont été formulées pour des services de type N-vers-1, pouvant ainsi déboucher sur un modèle approchant le multipoint-à-multipoint. Ces solutions sont en cours de discussions et pourront être intégrées au sein d'une prochaine version de PNNI et dans une extension de la signalisation UNI 4.0.

**Alternatives au multicast actuel** Au sein de la communauté scientifique, plusieurs travaux proposent la gestion du modèle *many-to-many* au niveau ATM. Ils sont généralement basés sur des arbres partagés et évitent l'entrelacement des cellules. On peut citer **SEAM** (*Scalable and Efficient ATM Multipoint-to-Multipoint Multicasting*) [GR97] de l'INRIA/ATT Research et **SMART** (*Shared Many-to-many ATM Reservation*) [GBO97] de l'EPFL.

Toutes ces approches ont été envisagées au niveau ATM, et nécessitent une coûteuse mise en œuvre. Pour la gestion du *multicast* dans les réseaux ATM, c'est-à-dire du service *many-to-many*, nous proposons dans le chapitre suivant une solution située dans la couche protocolaire supérieure.

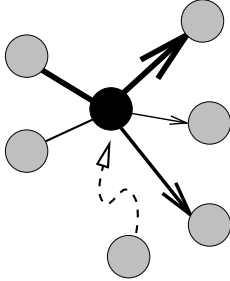
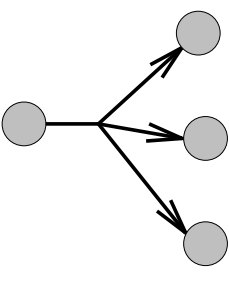
## 2.4 CONCLUSION DU CHAPITRE

Comme nous avons pu le constater au sein de ce chapitre, les réseaux IP et ATM proposent des approches des services multipoints avec QoS différentes. L'intégration des services proposés autour d'IP est primordiale dans les solutions futures, à cause de leur ubiquité au niveau des interfaces applicatives à terme. Quant aux services des réseaux ATM, ce sont actuellement les principaux disponibles pour supporter les besoins d'IP. Il est donc nécessaire pour ces deux technologies d'interfonctionner.

Les services IP, avec les extensions **IP Multicast** et **RSVP/IntServ** proposent les caractéristiques suivantes : initiation par les récepteurs, agrégation des flots, gestion dynamique de QoS variées sur les branches de l'arbre de diffusion.

Les services des réseaux ATM, en se basant sur l'interface UNI 3.x globalement disponible et proposant un service quasi-identique à celui de l'UNI 4.0 pour le service multipoint, possèdent les caractéristiques suivantes : initiation au niveau de l'émetteur, avec une QoS non renégociable, une spécification explicite des récepteurs multipoints et la même QoS sur chaque branche de l'arbre de diffusion.

Une comparaison des deux services multipoints est présentée dans le Tableau 2.6.

	<b>Modèle RSVP</b>	<b>Modèle ATM</b>
<i>Mode</i>	Non connecté	Connecté
<i>Multipoint</i>	Multipoint-à-Multipoint Orienté récepteur Abstraction de groupe	Point-à-Multipoint Orienté émetteur
<i>QoS</i>	Dynamique Hétérogène	Statique Homogène
		

**Tab. 2.6** Comparaison des modèles de services

De nombreuses approches orthogonales entre les technologies existent, l'utilisation conjointe de celles-ci est loin d'être triviale et va demander la mise en œuvre de solutions complexes pour gérer un interfonctionnement efficacement. C'est l'objet du chapitre suivant.

## CHAPITRE 3

# *L'intégration d'IP sur les réseaux ATM*

### Contenu du chapitre

*Ce chapitre présente les intégrations d'IP sur ATM à travers deux approches : la première conserve la signalisation orientée connexion usuelle des réseaux ATM (solutions de l'IETF et l'ATM Forum); la seconde utilise directement la commutation ATM avec la signalisation IP (commutation IP).*

*Ce chapitre est organisé de la manière suivante :*

- ⇒ Section 3.1 – Introduction du chapitre – p. 32.
- ⇒ Section 3.2 – Les solutions classiques de l'IETF – p. 32 : *Description de la solution Classical IP (CLIP), de l'extension MARS pour le multicast, de la gestion des shortcuts (NHRP) et d'IPv6.*
- ⇒ Section 3.3 – Les solutions de l'ATM Forum – p. 43 : *Description de la solution LAN Emulation (LANE) et de son extension MPOA.*
- ⇒ Section 3.4 – Les solutions basées sur la commutation IP – p. 51 : *Présentation des approches orientées flot (IP Switching, CSR et IPSO FACTO) et des approches orientées topologie (Tag Switching, ARIS et MPLS). Étude du multicast.*
- ⇒ Section 3.5 – Conclusion du chapitre – p. 66.

### 3.1 INTRODUCTION DU CHAPITRE

L'ubiquité du protocole IP et l'adoption de la technologie ATM dans le milieu des télécommunications ont favorisé l'émergence de solutions IP sur ATM. Nous présentons au sein de ce chapitre les principales approches proposées pour intégrer IP sur les réseaux ATM.

Dans le chapitre précédent, nous avons étudié les approches orthogonales suivies par IP et ATM. Celles-ci imposent des solutions complexes pour réaliser l'intégration des deux technologies, et les nombreux acteurs en jeux ont multiplié les propositions [CSV96].

Nous nous intéressons particulièrement à trois types d'approches. Les deux premières gèrent le mode non connecté d'IP sur le mode connecté d'ATM à l'aide de serveurs. Elles sont proposées par chacun des deux principaux organismes du monde ATM : *LAN Emulation* (LANE) est proposé au sein de l'ATM Forum ; *Classical IP* (CLIP) est le résultat des groupes de travail de l'IETF. La dernière approche consiste à supprimer le mode connecté d'ATM et à intégrer directement de la signalisation IP.

Dans la suite, nous détaillons les principes et les services fournis par les différentes solutions basées sur ces trois approches. Nous abordons celles-ci dans l'ordre chronologique d'apparition des différents standards s'y relatant. Nous débutons ainsi avec les solutions basées sur CLIP, de la proposition de base aux extensions pour le multipoint avec *Multicast Address Resolution Server* (MARS) et la connectivité directe liée au *Next Hop Resolution Protocol* (NHRP). Puis nous décrivons la solution basée sur LANE et son extension multiprotocole pour la couche réseau (MPOA). Nous terminons par les solutions basées directement sur de la signalisation IP, que nous appelons "Commutation IP".

### 3.2 LES SOLUTIONS CLASSIQUES DE L'IETF

L'IETF s'intéresse à tous les types de média permettant le support du protocole IP. Les réseaux ATM ne dérogent pas à cette règle et ont suscité de nombreux travaux au sein de groupes de travail spécifiques. Initialement, deux groupes ont développé des solutions pour ATM ou pour des réseaux avec des propriétés similaires : IPATM (*IP Over ATM*) et ROLC (*Routing Over Large Clouds*). Ces deux groupes ont ensuite été réunis au sein du groupe ION (*Internetworking Over NBMA*). Ce groupe propose donc des solutions pour les réseaux NBMA (*NonBroadcast Multiple Access*), réseaux maillés reposant sur des technologies où il n'y a pas de diffusion native, tels que les réseaux ATM.

CLIP est la principale solution soutenue par l'IETF pour l'utilisation d'IP sur les réseaux ATM intégrant la signalisation orientée connexion de l'ATM Forum. Une autre approche, n'utilisant pas la signalisation ATM Forum, a été récemment pro-

posée par l'IETF à travers le groupe MPLS *MultiProtocol Label Switching*. Cette approche est une des solutions basées sur la "Commutation IP" et est présentée dans la section 3.4.2.

### 3.2.1 Classical IP and ARP over ATM (CLIP)

L'approche CLIP reprend le concept fondamental de l'Internet selon lequel IP est un protocole de convergence. L'approche de l'IETF a donc été de considérer ATM comme une technologie de sous-réseaux<sup>1</sup> de manière à utiliser tout ce qui est déjà opérationnel sur IP. L'accès au sous-réseau ATM se fait donc à travers l'interface d'IP et la communication entre les sous-réseaux, qu'ils soient de technologie ATM ou autre, se fait par l'intermédiaire de routeurs.

L'IETF, à travers le groupe de travail IPATM, a donc proposé avec CLIP une solution en couche (*Overlay*) utilisée pour faire fonctionner IP sur différents types de couches liaisons. Cette approche repose sur deux points clés : l'encapsulation et la résolution d'adresses, que nous présentons dans la suite.

#### Encapsulation avec CLIP

Heinaneen a défini l'encapsulation standard de CLIP dans le RFC 1483 [Hei93]. Les paquets IP sont intégrés dans les CPCS PDU (*Common Part Convergence Sublayer Protocol Data Unit*) de l'AAL5 et utilisent deux formats d'encapsulation : soit "Null" avec intégration directe des paquets IP ; soit "LLC/SNAP" (*Logical Link Control / SubNetwork Access Protocol*) avec une identification du protocole pour permettre le multiplexage de différents types de paquets. La Figure 3.1 présente les différents niveaux d'encapsulation pour ce format.

La taille maximum des paquets (ou MTU pour *Maximum Transmission Unit*) est par défaut de 9180 octets, mais cette valeur est négociable dans la limite des 64 Ko de la taille maximum d'un paquet AAL5. La définition de la MTU se trouve dans le RFC 2225 [LH98] qui remplace le RFC 1626 [Atk94] sur lequel repose les implémentations.

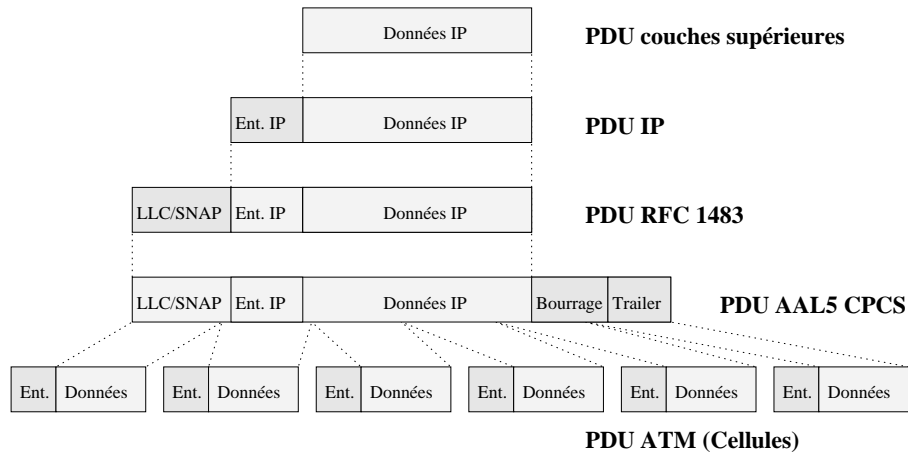
#### Résolution d'adresses avec CLIP

Pour effectuer la résolution d'adresses sur un médium sans diffusion (*broadcast*) tel que les réseaux ATM, l'utilisation d'un serveur est nécessaire. Le protocole ARP (*Address Resolution Protocol*) habituellement utilisé est remplacé par le protocole ATMARP. Celui-ci était initialement spécifié dans le RFC 1577 [Lau93] auquel se conforment les implémentations actuelles mais il a été rendu obsolète par le récent

---

<sup>1</sup>Dans l'environnement IP, un sous-réseau (*subnet*) est un réseau de communication connexe utilisant la même technologie.





**Fig. 3.1** Encapsulation CLIP

RFC 2225<sup>2</sup> [LH98]. Le client ATMARP réalise l'enregistrement auprès du serveur ATMARP et gère la résolution d'adresses avec celui-ci (qui est connue par configuration). Chaque membre s'inscrit donc auprès du serveur qui lui correspond en initiant un VC vers l'adresse ATM de celui-ci. Le serveur ATMARP apprend les adresses IP correspondantes à l'aide des informations contenues dans les requêtes qui lui sont adressées ou bien en utilisant le protocole de résolution inverse InARP (*Inverse ARP*) à partir de l'adresse ATM. Les réponses aux requêtes des clients sont identiques à celles que fournirait ARP.

Les limitations de l'approche Client/Serveur sont liées au nombre de clients. La technologie ATM fournit des réseaux à connectivité étendue, il faut donc segmenter les nuages ATM en plusieurs LIS (*Logical IP Subnetwork*). Les LIS sont des entités administratives disjointes qui, dans un réseau à diffusion, correspondraient à un lien (segment *Ethernet* par exemple). Un LIS est servi par un serveur ATMARP qui assure la correspondance entre les adresses ATM et les adresses IP. Pour les communications internes au LIS, les hôtes et routeurs peuvent établir des VC et communiquer directement. Les communications vers des hôtes externes au LIS utilisent des routeurs intermédiaires, même si un VC direct entre les deux hôtes peut être établi. La Figure 3.2 représente le chemin entre deux hôtes A et B appartenant à un même réseau ATM mais à des LIS différents. La communication est routée entre chaque LIS (routeurs R1 et R2).

### Conclusion sur CLIP

Le standard CLIP permet de résoudre simplement le problème de l'utilisation d'ATM avec IP en ne modifiant que légèrement la résolution d'adresse (ATMARP). Cette solution est adaptable à différentes tailles de réseau à condition de bien dimensionner

<sup>2</sup>Le RFC 2225 résulte de l'intégration et de la mise à jour des RFC 1577 et 1626

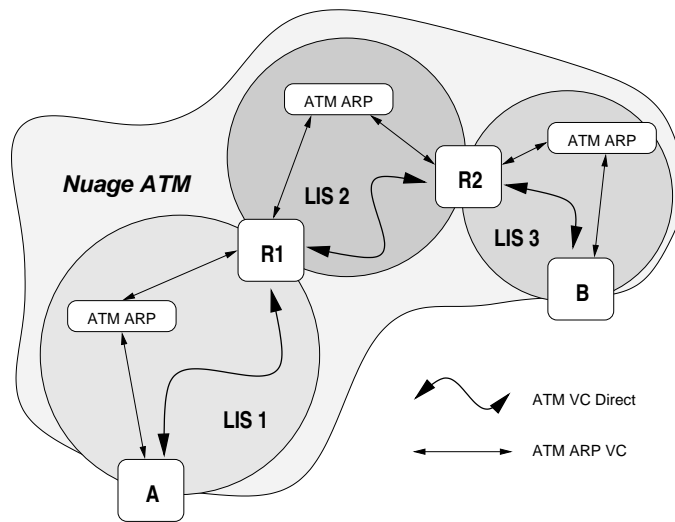


Fig. 3.2 Résolution d'adresses avec CLIP

les LIS. CLIP est spécifié pour le support des signalisations UNI 3.x [LMH<sup>+</sup>97] et UNI 4.0 [Mah98].

Plusieurs inconvénients sont à noter. Nous pouvons citer plus particulièrement :

**Client/Serveur** CLIP est basé sur un serveur unique centralisant les fonctions de résolution d'adresses. L'introduction de serveurs multiples est complexe.

**Mode connecté** CLIP repose sur le mode connecté fourni au niveau AAL5 par la signalisation ATM Forum. Des temps de latence importants sont à prévoir à l'initialisation des connexions.

**LAN Virtuels** L'utilisation des LIS est similaire à celle des LAN virtuels. La topologie des LIS est décorélée de l'architecture physique du réseau, ce qui est inefficace.

**Pas de Multicast** CLIP ne fournit que le service IP de base. Une extension multicast a été définie avec MARS (voir la section 3.2.2).

**Routeurs intermédiaires** L'utilisation de routeurs intermédiaires n'est pas efficace avec les réseaux à connectivité étendue. L'utilisation de VC inter LIS est proposée avec NHRP (voir la section 3.2.3).

**QoS** L'introduction de la QoS n'est pas abordée, étant donné que le service visé est celui d'IP de base. Nous proposons une solution avec RSVP dans le chapitre 4.

### 3.2.2 Multicast Address Resolution Server (MARS)

Le multicast IP propose une duplication des paquets vers les membres d'un groupe effectuée par le réseau (voir la section 2.2.2). Ce service a été initialement défini pour

des réseaux à diffusion et des liaisons point-à-point, où la gestion locale de ce mode de transmission est triviale (diffusion implicite à tous les membres du segment et sélection au niveau du récepteur). Sur les réseaux NBMA, il n'a pas de fonction de diffusion native. ATM ne dispose que d'un service de VC point-à-multipoint initialisés explicitement vers chaque récepteur concerné et sans abstraction de groupe. La solution CLIP ne fournissant que le service IP de base, Armitage a proposé une solution pour intégrer le multicast dans les réseaux sans diffusion [Arm97a]. Pour pallier les absences de fonctionnalités, de la même manière que pour CLIP avec ATMARP, la mise en place d'un serveur est nécessaire : MARS (*Multicast Address Resolution Server*) va gérer l'enregistrement et la distribution des adresses de groupe. Comme pour les LIS, il prend place dans une subdivision du réseau ATM appelée un *cluster*.

### Principes de MARS

Le standard relatif à MARS [Arm96] indique la manière d'implémenter le multicast IP [Dee89] sur un service ATM local tel que CLIP. Trois aspects principaux sont traités dans la spécification :

- la résolution d'adresses multicast en adresses ATM ;
- la gestion de groupe ;
- la transmission des données multicast.

En fait, MARS gère les adresses de groupe de manière générique, pour tout protocole de réseaux utilisant ATM, et pas seulement pour IP.

MARS peut être considéré comme une évolution du serveur ATMARP. Celui-ci utilisait une table de conversion des adresses IP pour tous les membres du LIS :

$$\boxed{\{ \text{Adresse-IP} \iff \text{Adresse-ATM} \}}$$

MARS étend la table aux adresses de groupe au sein d'un *cluster* et généralise l'adresse d'entrée :

$$\boxed{\{ \text{Adresse-de-niveau-3} \iff \text{Adr-ATM.1, Adr-ATM.2, ... Adr-ATM.n} \}}$$

MARS peut résider dans tout élément du réseau directement accessible par les utilisateurs de ce serveur. Les interfaces qui veulent se joindre à un groupe doivent connaître l'adresse où leur MARS réside. MARS n'est qu'un serveur de résolution d'adresses, il reste à créer les VC ATM point-à-multipoint pour obtenir un service multicast. A l'intérieur d'un *cluster*, deux modèles permettent ce service : le **maillage multicast** et le **serveur multicast**.

### Maillage multicast avec MARS

Le premier modèle utilisé pour transmettre des données consiste à faire un maillage multicast (présentée dans la section 2.3.3). Le maillage est la solution la plus simple à mettre en œuvre et est utilisé par défaut. Un client s'enregistre auprès de son MARS après avoir créé un VC point-à-point pour la signalisation (voir figure 3.3). Le serveur MARS ajoute une branche à son VC point-à-multipoint de contrôle (*ClusterControlVC*) de manière à communiquer les arrivées ou les départs de membres d'un groupe donné à tous ses clients simultanément. Le client peut ensuite envoyer une requête demandant les membres d'un groupe multicast et recevoir la liste des adresses ATM des membres de ce groupe.

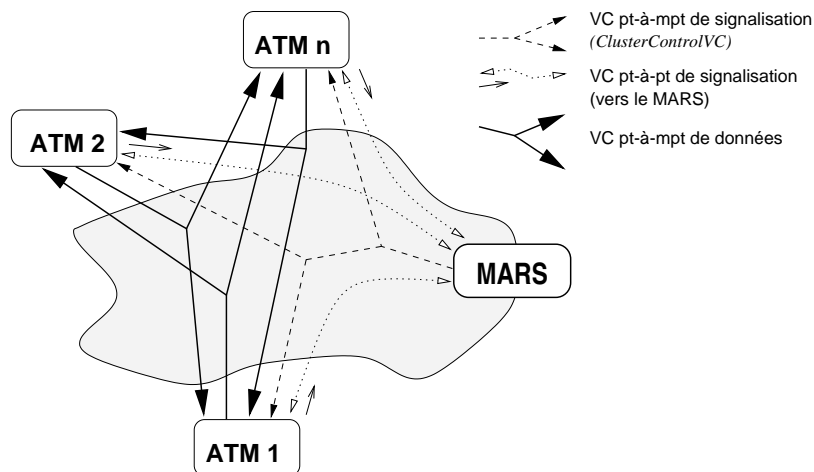


Fig. 3.3 Maillage multicast avec MARS

Avec le maillage multicast, une fois les adresses résolues, des VC point-à-multipoint sont établis par chaque source. Les connexions sont donc gérées localement et mises à jour par les messages provenant du *ClusterControlVC*.

Ce modèle est évidemment utilisé dans des configurations légères, car la charge de la gestion des connexions par les sources va rapidement croître avec le nombre de récepteurs dans le groupe.

### Serveur multicast avec MARS

L'évolution du maillage multicast est l'utilisation d'un serveur intermédiaire appelé "serveur multicast" ou MCS (*MultiCast Server*), dont le principe est basé sur un serveur *proxy* pour la création des VC point-à-multipoint (présenté dans la section 2.3.3). Ce modèle repose sur un schéma où chaque source établit un VC point-à-point vers un MCS pour lui envoyer son trafic multicast, et le serveur utilise son VC point-à-multipoint pour retransmettre ce trafic.

Les clients s'enregistrent de manière identique, mais lorsqu'ils demandent les adresses ATM associées à un groupe multicast, ils reçoivent la ou les adresses correspondant aux MCS gérant ce groupe. Au niveau de la signalisation, le serveur MARS rajoute un autre VC point-à-multipoint de contrôle, le *ServerControl VC*, de manière à communiquer les arrivées ou départs de membres d'un groupe donné aux MCS.

La Figure 3.4 présente un exemple avec deux MCS contrôlés par un MARS dans un cluster. Cette configuration peut correspondre à l'association d'un groupe multicast par MCS. Les émetteurs vers un groupe multicast ont un VC point-à-point vers le MCS correspondant et les membres du groupe sont les feuilles des arbres multipoints partant de ces MCS. Sur la figure, le MCS 1 transmet aux membres ATM 1 et ATM 2 alors que le MCS 2 envoie des données à ATM 2 et ATM n.

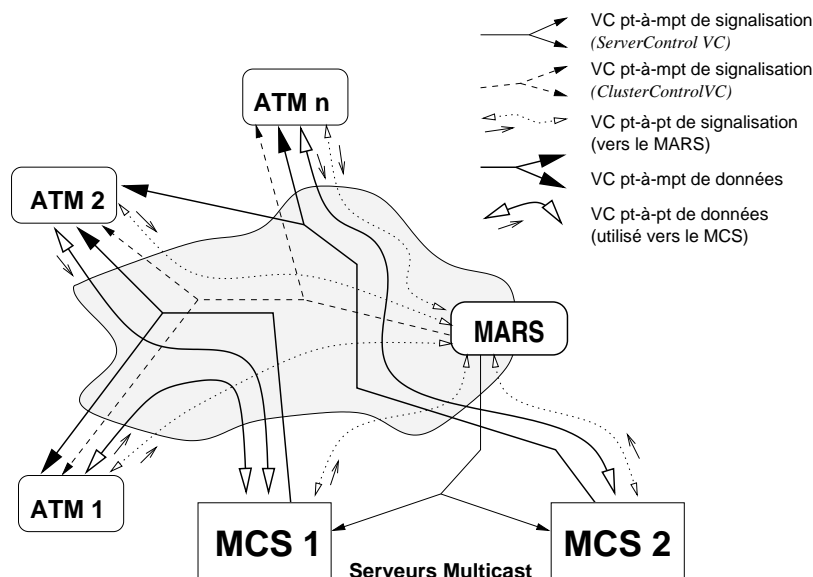


Fig. 3.4 Serveur multicast avec MARS

Même si la spécification de MARS explique les nombreuses interactions entre le MCS et MARS, tant au niveau des mécanismes liés à la fiabilité qu'à celui de la migration des membres vers différents MCS, peu d'informations sont disponibles pour la mise en œuvre de ces MCS. Talpade et Amar ont proposé une étude détaillée [TA97] en vue de l'implémentation des MCS dans un environnement multicast basé sur MARS.

### Exemple de transmission avec MARS

Dans la Figure 3.5, nous proposons un exemple synthétisant les différentes topologies et les divers mécanismes utilisés lors d'une transmission multicast au sein d'un réseau ATM composé de plusieurs clusters. La configuration comporte trois clusters qui sont reliés à l'aide de trois **m**routers utilisant si besoin un tunnel IP multicast

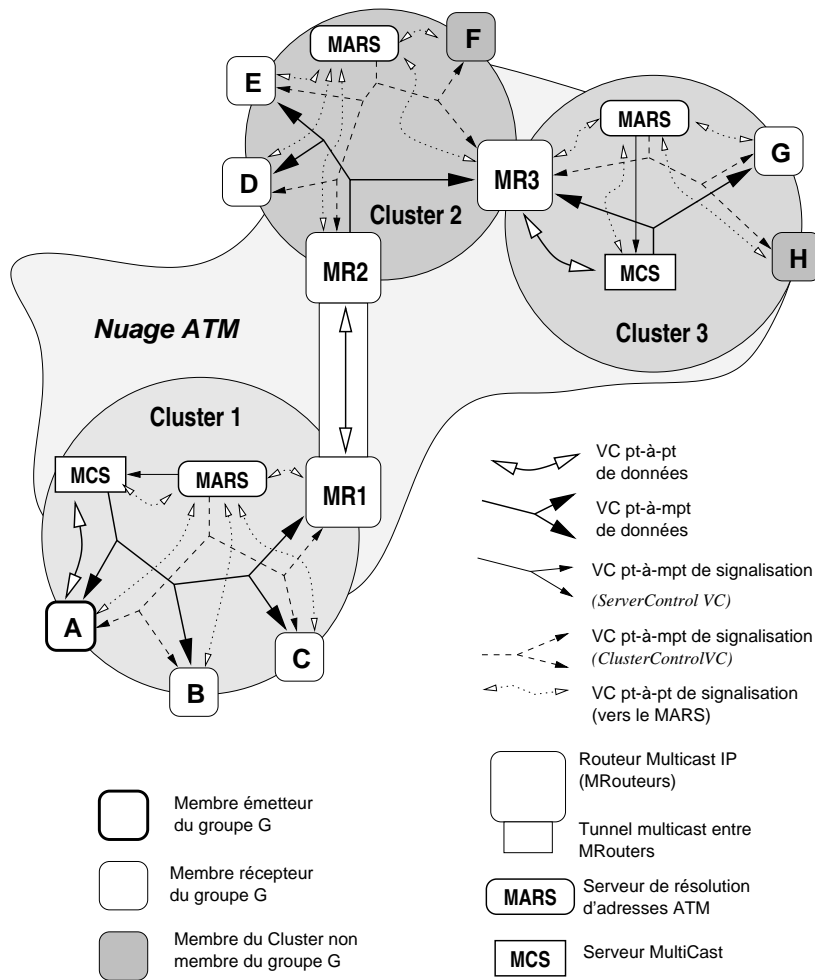


Fig. 3.5 Transmission vers un groupe multi-cluster

(reposant sur une technologie non spécifiée). Les Cluster 1 et Cluster 3 reposent sur un modèle avec MCS alors que le Cluster 2 est basé sur un maillage multicast. Nous considérons la transmission au sein d'un groupe unique, avec un émetteur A et des récepteurs répartis sur les trois clusters. Nous supposons une situation initiale avec seulement les membres A, B, D et G abonnés au groupe. Les mrouteurs reçoivent toutes les communications multipoints, ils sont donc membres de tous les arbres de distribution.

Les membres A, B et G sont irrigués par un MCS. Ils ont établi les VC de contrôle vers le MARS et s'y sont enregistrés. Le MARS a rajouté des branches au *ClusterControlVC* vers ces membres et a informé le MCS à travers le *ServeurControlVC* pour qu'il rajoute les branches nécessaires vers ces mêmes membres au VC point-à-multipoint partagé. Le membre D appartient à un cluster qui utilise un maillage multicast. Après avoir établi un VC vers le MARS, celui-ci obtient les adresses des hôtes vers lesquels il doit initier un VC point-à-multipoint pour les données.

L'émetteur A commence la transmission. Il initie un VC point-à-point vers le MCS de son cluster, celui-ci relaie les données vers les membres du groupe du Cluster 1 et vers le MRouter 1. Le mrouter encapsule les données pour les transmettre au MRouter 2 du Cluster 2. Celui-ci gère un VC point-à-multipoint vers les membres locaux et le MRouter 3. Ce dernier est directement connecté au Cluster 3 et va donc diffuser à son MCS qui va envoyer les données à G.

Pendant l'émission de A, nous intégrons de nouveaux membres au groupe. Au sein du Cluster 1, le membre C rejoint le groupe. Son enregistrement auprès du MARS génère un message sur le *ServerControlVC* pour indiquer la modification à effectuer au niveau VC point-à-multipoint partagé. Pour le Cluster 2, le membre E rejoint le groupe, son enregistrement auprès du MARS génère un message sur le *ClusterControlVC* qui indique à tous les membres locaux la modification à apporter à leur liste et à leur VC point-à-multipoint.

### Autres propositions avec le multicast

En extension ou en complément de MARS, plusieurs solutions ont été proposées, avec pour but principal l'extension au delà des limites du cluster de la gestion du multicast :

**VENUS** La proposition *Very Extensive Non Unicast Service* [Arm97b] introduit les concepts d'un service multicast étendu pour permettre des communications inter-cluster efficaces.

**EARTH** La proposition *EAsy IP multicast Routing THrough ATM clouds* [Smi97] ne subdivise pas le réseau ATM pour la résolution multicast. Les interfaces multicast sont associées à un LIS de la taille du nuage ATM appelé MLIS (*Multicast Logical IP Subnet*) et établissent une connexion de contrôle vers le serveur EARTH qui gère la résolution des adresses IP multicast en adresse

ATM. Cette approche permet une simplification mais introduit de nombreuses limitations.

### 3.2.3 Next Hop Resolution Protocol (NHRP)

Le protocole NHRP est une technique proposée par le groupe de travail ROLC (*Routing Over Large Clouds*) pour la résolution d'adresses au sein des réseaux NBMA. Ce protocole est spécifié dans [LKP<sup>+</sup>98].

Nous avons vu précédemment qu'il était nécessaire, avec CLIP, de subdiviser le nuage ATM en plusieurs LIS. ATMARP fournit la correspondance entre adresses ATM et IP au niveau des membres du LIS. En utilisant NHRP à la place de ATMARP, on peut connaître l'adresse ATM d'un membre distant pour établir un VC direct, appelé *shortcut*, et ainsi éviter le passage à travers des routeurs intermédiaires. Ce protocole permet donc la résolution d'adresses de destinataires directement accessibles au sein du réseau sous-jacent mais pas au niveau IP.

Cette solution semble séduisante mais le gain de performance obtenu en évitant les routeurs et les remontées protocolaires est à pondérer par le bris de la hiérarchie de routage (le nouveau chemin ne respecte pas le routage IP), entraînant de nouveaux problèmes liés aux boucles et routages asymétriques. Cette approche est surtout intéressante pour les flots à longue durée car la résolution d'adresses nécessite plusieurs requêtes contrairement à l'approche ATMARP.

#### Principes de NHRP

NHRP repose sur un ensemble de serveurs NHS (*Next Hop Server*) coopératifs. Dans chaque sous-réseau, un NHS maintient une table de résolution d'adresses IP et ATM. Le NHS a autorité sur ce domaine car il possède la table de référence.

La résolution d'une requête NHRP est effectuée lorsqu'un NHC (*Next Hop Client*) tente d'établir une connexion vers un hôte dont il ne connaît pas l'adresse ATM. Un NHS vérifie s'il a autorité sur l'adresse à résoudre. Si oui, il y répond, sinon il peut envoyer une adresse qu'il a dans son cache ou transmettre à un autre NHS selon le protocole de routage d'IP. La Figure 3.6 présente la succession de requêtes.

Lorsque le destinataire est hors de la zone de résolution de NHRP, le protocole renvoie l'adresse ATM du routeur le plus proche. Pendant la résolution d'adresses NHRP, les données peuvent commencer par utiliser le chemin routé conventionnel avant de passer par le *shortcuts*.

#### Raccourcis et Multicast

L'utilisation de raccourcis (*shortcuts*) au lieu du routage classique dépend du service que l'on demande au réseau [Can98]. Le multicast est lié au niveau IP et est efficace



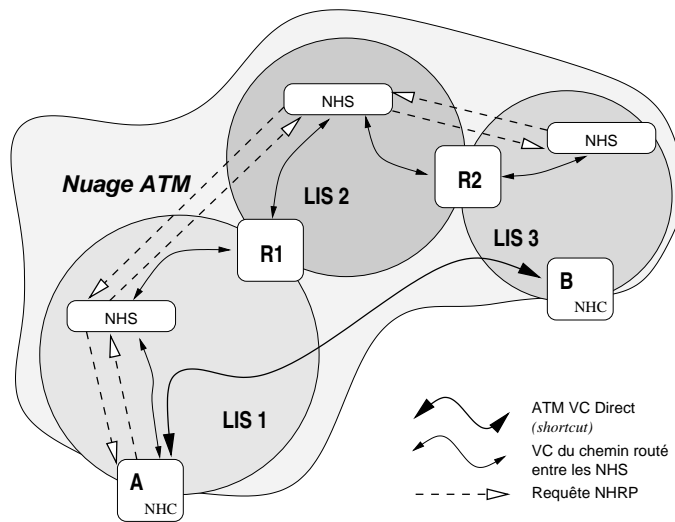


Fig. 3.6 Les raccourcis de NHRP

si l'on utilise de nombreux éléments intermédiaires au niveau IP. Les problèmes vis-à-vis du multicast ne sont pas facilement gérables avec NHRP. Il est nécessaire d'introduire une résolution d'adresses complexe et initiée par l'émetteur pour chaque membre d'un groupe. De plus, un multicast efficace demande la connaissance de la topologie du réseau pour que IP puisse créer des arbres avec une duplication minimum. L'approche basée sur les *shortcut* est donc en opposition avec celle du *multicast*.

### 3.2.4 IPv6 over ATM

Les solutions précédentes ont été développées pour IPv4. Des travaux d'adaptation de ces modèles ou de nouveaux pour IPv6 sont en cours. Le transfert au niveau IP de la fonctionnalité de résolution d'adresses ARP ne permet plus d'utiliser ATMARP. La résolution d'adresses étant effectuée par ND dans IPv6, il faut donc étudier l'adaptation de celui-ci sur ATM.

Le groupe de travail ION de l'IETF propose une solution [ASJH98a] permettant l'utilisation de IPv6 sur les réseaux NBMA (*NonBroadcast Multiple Access*) dont les réseaux ATM font partie. Un document spécifique [ASJH98b] indique comment utiliser cette solution dans le cas d'ATM. Le principe utilisé repose sur l'extension de la notion de lien d'IPv6, appelé "Lien Virtuel" (*Logical Link*), et qui correspond au LIS de CLIP. Le *multicast* au sein du Lien Virtuel est géré par MARS et les *shortcut* sont intégrés grâce à NHRP.

### 3.2.5 Perspectives

L'approche CLIP de l'IETF ne propose pas de solutions simples pour obtenir le service IP sur les réseaux ATM. Cette complexité est principalement due au mode orienté connexion d'ATM et liée à l'introduction nécessaire de nombreux serveurs.

L'utilisation de solutions multiples pour résoudre divers problèmes liés aux réseaux NBMA indique que le modèle utilisé est une solution transitoire avant l'apparition d'une solution globale plus complète et efficace.

Malgré la complexité de CLIP, ce modèle qui a été le premier disponible (1993), fonctionne correctement et est largement disponible. Il mérite à ce titre des développements. Nous présenterons au chapitre suivant l'utilisation de ce service multipoint avec QoS basé sur CLIP.

Parallèlement à CLIP, due à l'IETF, une autre solution s'est développée à l'ATM Forum. Elle repose sur des concepts de base identiques mais propose un service de niveau MAC. Nous la détaillons maintenant.

## 3.3 LES SOLUTIONS DE L'ATM FORUM

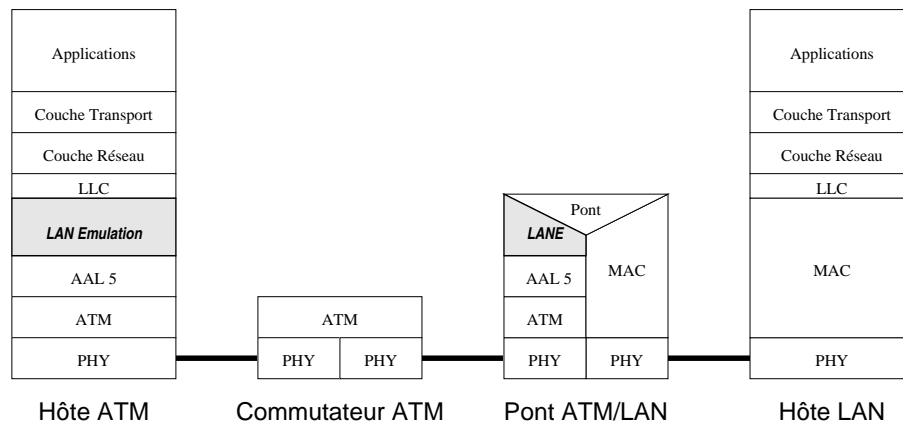
Le but majeur de l'ATM Forum est de permettre l'utilisation de la technologie ATM aux réseaux existants. ATM doit apparaître tel un média de transmission parmi d'autres. La couche d'adaptation choisie est la couche MAC des réseaux locaux de type IEEE 802.

Nous présentons la solution ATM Forum de base pour le trafic de données, appelée *LAN Emulation* (LANE), puis nous décrivons la solution proposée pour l'interconnexion de sous-réseaux, nommé MPOA.

### 3.3.1 LAN Emulation

Le but de LANE est de fournir un service MAC au dessus d'ATM à tout ce qui fonctionne sur le service MAC d'*Ethernet* (IEEE 802.3) ou de *Token Ring* (IEEE 802.5), en échange de quelques sacrifices au niveau des fonctionnalités ATM (les services attendus sont ceux des réseaux locaux et non pas ceux d'ATM).

Le principe est donc similaire à CLIP mais avec une interface au niveau MAC (voir la Figure 3.7), de manière à ré-utiliser sans modification les différents protocoles de niveau réseau (tels qu'IP) et donc les applications courantes. Ainsi chaque composant possédera une adresse MAC en plus de l'adresse ATM, ce qui n'est pas le cas avec CLIP. LANE version 1 [LAN95] a été défini en 1995. Une présentation détaillée de LANE se trouve dans l'article [FM96].



**Fig. 3.7** La couche LAN Emulation

## Principes de LANE

Un ELAN (*Emulated Local Area Network*) est défini par un groupe de stations ou de ponts connectés directement à un réseau ATM. Le concept de ELAN correspond à la notion de sous-réseau IP.

Le problème principal à résoudre provient du fait qu'ATM est un réseau maillé sans diffusion (NBMA) alors que le service à fournir est adapté à des média utilisant la diffusion. La mise en œuvre de fonctionnalités la remplaçant, à l'aide d'un modèle basé sur des serveurs, est donc nécessaire. Comme pour CLIP, le service à fournir repose sur le mode sans connexion. La création de VC entre deux entités d'un même ELAN est effectuée avant toute transmission de données après une résolution d'adresses MAC/ATM. L'encapsulation des données est réalisée sur l'AAL 5.

L'architecture de base de LANE repose sur un client et 3 serveurs :

**LEC** Le *LAN Emulation Client* est la partie client et se situe au sein de l'hôte ou d'un pont (élément interfaçant des réseaux locaux classiques avec le réseau ATM). Il est utilisé principalement pour le transfert des données et la résolution d'adresses. Il assure aussi des fonctions de gestion en contactant les serveurs explicitement pour s'y enregistrer.

**LECS** Le *LAN Emulation Configuration Server* permet la configuration des LEC (il indique l'adresse du LES par exemple). Il possède une adresse ATM connue.

**LES** Le *LAN Emulation Server* maintient une table de correspondance des adresses MAC et ATM des LEC enregistrées dans son LANE, pour effectuer les résolutions d'adresses.

**BUS** Le *Broadcast and Unknown Server* gère les services de diffusion (*broadcast*) ainsi que le trafic multicast. Il permet aussi de diffuser le trafic unicast dont l'adresse destination n'est pas résolue.

Il y a au sein d'un ELAN plusieurs LEC et un ou plusieurs serveur de chacun des LECS, LES et BUS.

Dans la seconde version de la spécification de LANE, les relations entre LEC et serveurs sont définies dans LUNI (*LAN Emulation User-Network Interface*) [LAN97b] et les relations entre serveurs dans LNNI (*LAN Emulation Network-Network Interface*) [LAN97a]. La spécification LNNI n'est pas encore standardisée.

Les principes généraux restent identiques pour les deux versions de LANE. Le changement qui nous intéresse dans LANEv2 est l'introduction du multicast (seule la diffusion est disponible dans LANEv1).

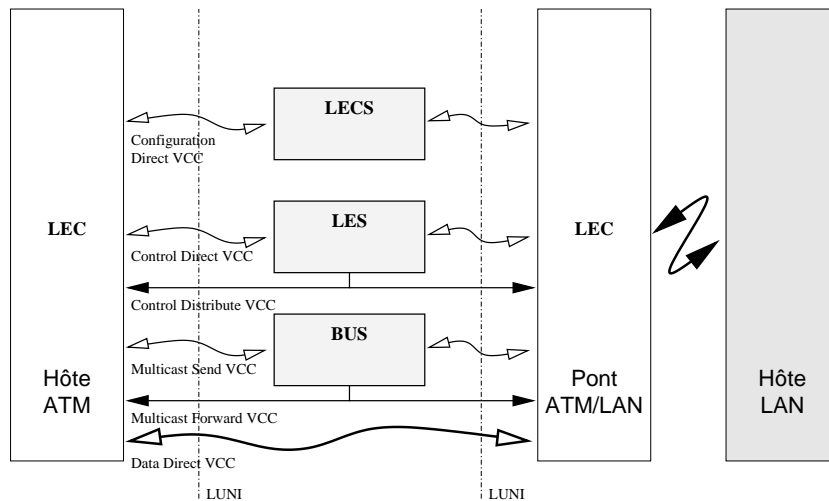


Fig. 3.8 Les différentes connexions de LANE

Les connexions établies au sein d'un ELAN entre clients et serveurs à travers l'interface LUNI, sont représentées dans la Figure 3.8 et sont les suivantes :

- Des connexions de contrôle, où l'on trouve : le *Configuration Direct VCC*, liaison point-à-point qui lie le LEC et le LECS ; le *Control Direct VCC*, liaison point-à-point qui lie le LEC au LES ; le *Control Distribute VCC*, liaison point-à-multipoint qui lie le LES vers les LEC.
- Des connexions pour les données : le *Data Direct VCC*, liaison point-à-point entre deux LEC ; le *Multicast Send VCC*, liaison point-à-point qui lie le LEC au BUS ; le *Multicast Forward VCC*, liaison point-à-multipoint qui lie le BUS vers les LEC.

Pour établir une communication, un LEC doit suivre plusieurs étapes. Une connexion initiale est établie avec le LECS pour obtenir les informations de configuration, dont l'adresse du LES. Puis les connexions avec le LES sont établies et l'adresse MAC du LEC est enregistrée. Pour finir, la création des connexions avec le BUS est réalisée à l'aide d'une résolution d'adresses broadcast.

Une fois l'initialisation terminée, le LEC peut créer des VC vers d'autres LEC de son ELAN, après avoir effectué une résolution d'adresses. Celle-ci est réalisée grâce à une requête de résolution d'adresses au LES. La source peut déjà envoyer des données à travers le BUS qui les diffusera. Au retour de la requête, le VC direct sera établi.

### Communication réseau avec LANE

Même si LANE n'a pas les limitations topologiques des réseaux locaux, elle reste une solution de sous-réseaux et est limitée en nombre de stations associées à un ELAN. Les membres de différents ELAN dans un réseau ATM doivent passer par des routeurs intermédiaires même si la connectivité ATM directe est possible. Ce passage risque d'être un goulot d'étranglement vu les débits disponibles avec les interfaces ATM.

### Communication multicast LANE

Pour réaliser la diffusion et le multicast dans un environnement ATM, l'approche "serveur multicast" vue précédemment (voir la section 2.3.3) est utilisée. Le serveur utilisé pour pallier l'absence de ces fonctionnalités dans les réseaux ATM est le BUS.

LANEv1 utilise uniquement la diffusion pour assurer le service multicast. Le principe d'inondation (*flooding*) puis de filtrage au niveau du LEC est utilisé pour le multicast (voir la Figure 3.9). Cette approche est peu efficace au sein d'un réseau maillé.

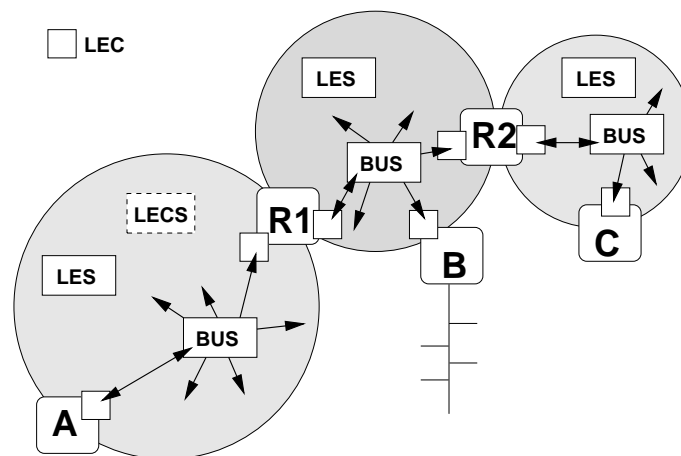


Fig. 3.9 Communication multicast avec LANE

LANEv2 assure la séparation du trafic de diffusion et du trafic multicast. Des VC multicast spécifiques sont utilisés et la gestion des adresses multicast est intégrée au LES.

### 3.3.2 MultiProtocol Over ATM

Après la solution LANE pour un service de niveau liaison, l'ATM Forum a proposé une solution fournissant un service de bout-en-bout au niveau réseau avec MPOA (*MultiProtocol Over ATM*). Cette solution propose :

- Une connectivité de **niveau 3** et ce théoriquement pour plusieurs types de protocoles réseau même si IP est principalement visé.
- Un accès aux hôtes directement connectés à ATM (avec une pile protocolaire MPOA) ou aux hôtes avec un accès classique à travers des éléments de pontage appelés *edge devices*.
- Une transmission à travers le réseau ATM basée soit sur le **routing** classique, soit sur des **liaisons directes** (*shortcuts*) à l'aide du protocole NHRP de l'IETF.
- L'intégration avec le standard LANE et l'interfonctionnement avec d'autres solutions de niveau 3 sur ATM (CLIP).
- La gestion du **multicast** (et du *broadcast*), à l'aide de la proposition MARS de l'IETF, ainsi que de la **QoS**.
- Un **routing virtuel** avec séparation des fonctions de relayage (*forwarding*) et calcul de route.

MPOA repose sur une subdivision du nuage ATM en des groupes d'hôtes basée sur les adresses de niveau 3 et appelée IASG (*Internetwork Address Sub-Group*). Cette division correspond à la notion d'adresse de sous-réseau d'IP.

La proposition MPOA initiale étant très ambitieuse, il en résulta des spécifications préliminaires d'une complexité exagérée. Pour finaliser une première version, le nombre de fonctionnalités proposées a été revu à la baisse, évacuant ainsi les problèmes non résolus pour l'heure, tels que le multicast (difficile à utiliser avec les *shortcuts*) ou l'intégration de la QoS.

La spécification de MPOA version 1 conserve le concept de communication de bout-en-bout au niveau 3 avec deux modes de connectivité à travers le **routing** et les **shortcuts**. La Figure 3.10 présente les possibilités de transmission de MPOA de A, un hôte directement connecté, à B, un *edge devices* (commutateur ATM de périphérie assurant les fonctions de pont et accédant à un réseau local classique). Le chemin routé passant par des serveurs MPOA (MPS) est représenté par des traits continus, et le chemin direct, évitant les éléments routants, en pointillé.

Le standard MPOA version 1 a été approuvé en juillet 1997 [MPO97]. Les fonctionnalités actuelles font de cette version une extension du standard LANE (on assimile un IASG à un ELAN). C'est pourquoi il a été décidé au sein de l'ATM Forum de fusionner les groupes de travail MPOA et LANE en un seul et même groupe.

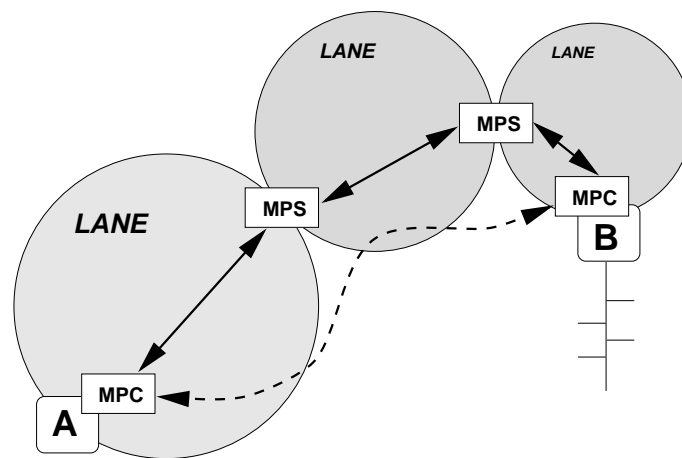


Fig. 3.10 Utilisation d'un *shortcut* MPOA

## Principes de MPOA

Les mécanismes principaux de MPOA résultent de l'utilisation de LANE comme protocole de base, ainsi que de l'intégration de la résolution d'adresses en dehors du ELAN à l'aide de NHRP.

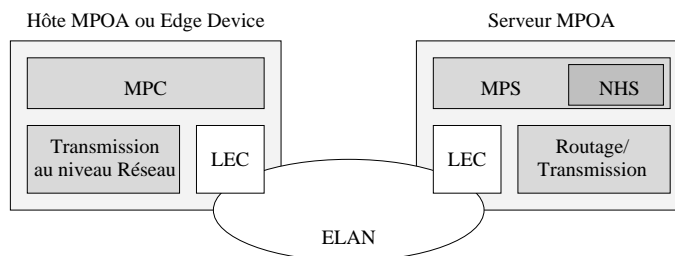


Fig. 3.11 Les composants MPOA

Ces mécanismes reposent sur les deux composants de base de l'architecture MPOA représentés dans la Figure 3.11 :

**MPC** (*MultiProtocol Client*) La partie client de MPOA s'intègre au sein des hôtes ATM ou des ponts (*edge devices*) en complément du LEC et de la gestion des fonctionnalités de niveau réseau. Le MPC a pour charge d'émettre des requêtes vers le MPS pour la résolution d'adresses ATM en dehors de son ELAN.

**MPS** (*MultiProtocol Server*) La partie serveur de MPOA est hébergée dans un routeur ou un serveur de route qui intègre un LEC. Le MPS inclue un NHS (*Next Hop Server*) pour permettre la résolution d'adresses distantes à l'aide du protocole NHRP.

Les communications entre MPC et MPS sont réalisées à l'aide de LANE. MPOA repose intégralement sur ce protocole à qui il fournit une résolution d'adresses éten-

due au niveau réseau. La Figure 3.12 présente les extensions introduites au modèle LANE de la Figure 3.8 pour utiliser MPOA.

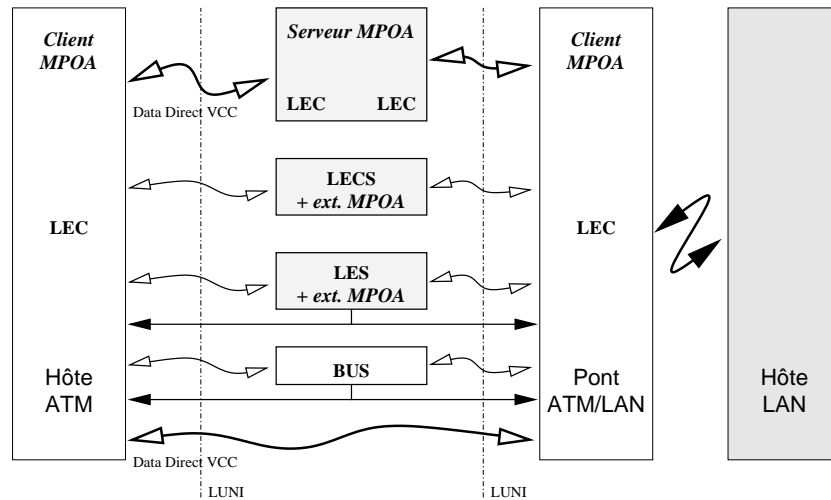


Fig. 3.12 Les connexions MPOA

MPOA repose sur le standard LANEv2 et accède aux différents éléments du service à travers LUNI. Le LECS et le LES sont modifiés et étendus pour transmettre les informations relatives à MPOA (configuration et enregistrement), de manière à établir un VC direct entre MPC et MPS qui se trouvent au sein du même LANE. Une fois le lien établi, la transmission de données peut être réalisée de manière classique, en considérant le MPS comme un élément intermédiaire au niveau réseau. Pour réaliser un *shortcut*, l'utilisation de NHRP est nécessaire.

NHRP est utilisé pour permettre la résolution d'adresses d'un hôte ou pont MPOA éloigné. La résolution d'adresses est gérée au niveau local par MPOA, et au niveau du réseau par NHRP comme cela est représenté sur la Figure 3.13. A partir du MPS entrant, situé dans le ELAN de l'émetteur, jusqu'au MPS sortant, situé dans le ELAN du destinataire, des requêtes NHRP sont échangées entre NHS.

Il en résulte pour l'établissement d'un *shortcut* de nombreux échanges de messages. La Figure 3.14 les présente. Pour accéder à un MPC sortant, le MPC entrant établit une connexion vers son MPS à l'aide du protocole LANE. Puis le MPC envoie au MPS entrant une requête MPOA. Cette requête est convertie au niveau NHRP et transmise au MPS sortant. Le MPS sortant entame alors une double conversion d'adresse (adresse réseau vers adresse MAC puis vers adresse ATM) et un échange IC (*Cache Imposition*) pour permettre l'établissement du *shortcut*. Pour finir, la réponse attendue transite par le chemin inverse pour rejoindre le MPC entrant.



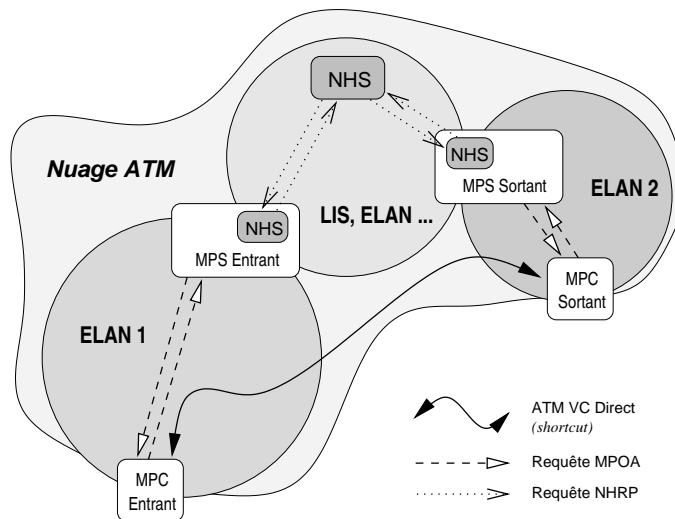


Fig. 3.13 Résolution d'adresses étendue avec NHRP dans MPOA

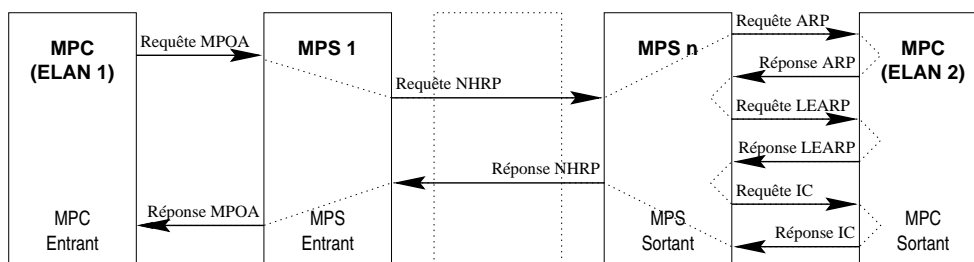


Fig. 3.14 Création d'un *Shortcut* MPOA

### 3.3.3 Perspectives

Le défaut principal de LANE et de MPOA est la complexité. L'importance du code nécessaire à l'implémentation de cette solution est révélateur : alors que le contrôleur d'une carte Ethernet classique comporte environ 2.000 lignes de code, pour fournir un service équivalent, la signalisation ATM avec LANE comprend plus de 60.000 lignes. MPOA nous évite le passage par les routeurs mais augmente encore considérablement la taille du code (200.000 lignes annoncées). Cette croissance considérable de la complexité est due au nombreux mécanismes mis en œuvre pour réaliser la gestion du mode connecté d'ATM (introduction de nombreux serveurs) et de l'abstraction de la topologie (ELAN ou IASG indépendants de la structure physique du réseau).

Les avantages liés à l'aspect multiprotocolaire ne nous intéressent pas car nous pensons qu'IP est l'interface universelle à terme. De plus, pour MPOA, le *multiprotocol* est pour le moment limité au seul IPv4.

Au niveau du multicast, son intégration est dans LANEv2 mais pas encore dans MPOA où son introduction au niveau réseau est beaucoup plus complexe, particulièrement avec les *shortcut*. Le multicast prévu initialement est de type MARS, ce qui ne résout pas les problèmes rencontrés précédemment.

L'approche au niveau MAC ne permet pas un accès aux paramètres de la QoS ATM. La QoS n'est pas abordée dans la spécification de MPOA, mais elle devrait être similaire à celle prévue pour LANEv2, et reposer sur une gestion administrative locale entre deux hôtes.

Les défauts des solutions que nous avons présentées reposent principalement sur le mode connecté utilisé par ATM. Des propositions ont abordé le problème différemment, évitant ainsi l'utilisation de serveurs et proposant une utilisation plus directe des capacités ATM. Nous les présentons maintenant.

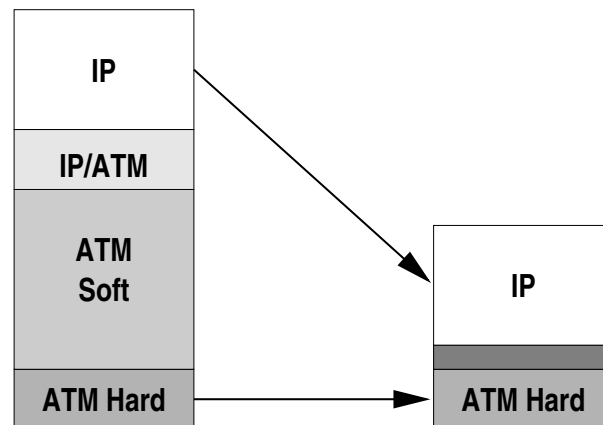
## 3.4 LES SOLUTIONS BASÉES SUR LA COMMUTATION IP

Le point commun des approches présentées précédemment est la conservation du mode connecté fourni par le service ATM. Cela implique une mise en œuvre protocolaire importante et complexe qui nuit à l'utilisation des performances de la technologie ATM.

Schématiquement, en se situant dans un modèle protocolaire en couche, l'approche basée sur le recouvrement (*overlaying*) est représentée avec le service IP et sa signalisation dans la couche supérieure, tandis que le service ATM et la commutation associée se situent dans la couche inférieure. Entre les deux se trouvent plusieurs couches faisant appel à de nombreux protocoles pour faire interfonctionner les deux précédentes : une première couche d'adaptation fournit un service orienté connexion (UNI) au dessus de la commutation de cellules (apparentée au mode non connecté)

puis différentes autres couches adaptent ce service en mode connecté pour une utilisation avec le mode non connecté utilisé par IP, à travers une solution telle que CLIP ou LANE.

Pour simplifier cette architecture, plusieurs groupes de travail ont choisi de minimiser les couches intermédiaires entre la signalisation IP et la commutation ATM, et particulièrement le passage en mode connecté. Les propositions résultantes intègrent une commutation directe au niveau ATM du trafic de données, en conservant un niveau d'adaptation minimum. Cette technique est appelée **commutation IP**. Le principe est représenté dans la Figure 3.15.

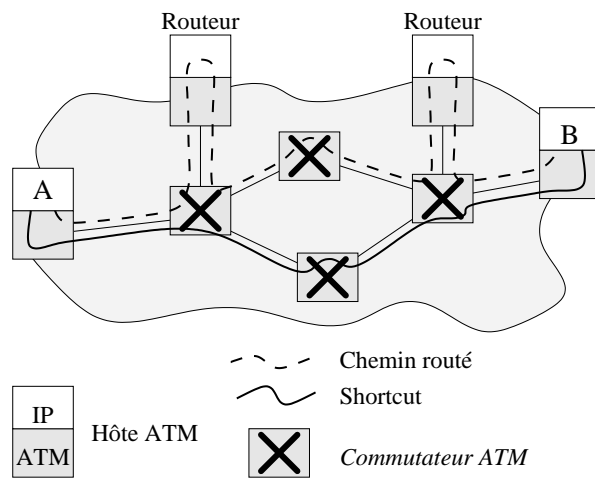


**Fig. 3.15** Moins de couches intermédiaires !

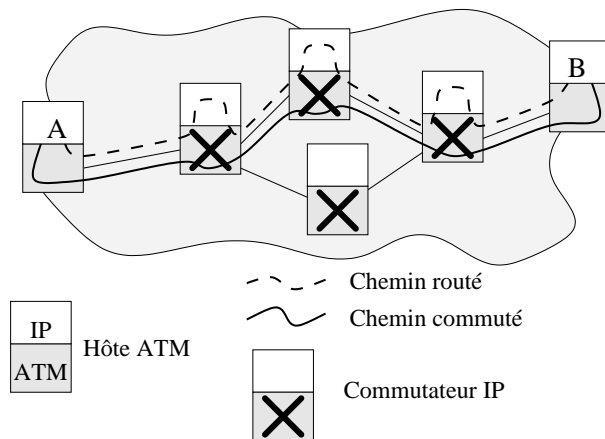
Les solutions IP sur ATM précédentes sont, du point de vue des couches protocolaires, la superposition d'un service de bout-en-bout de niveau réseau (le service IP) sur un autre service de bout-en-bout (le service ATM lorsque l'on se situe dans un nuage ATM). Mais le service ATM est ici assimilé à un niveau liaison selon le principe de recouvrement d'IP et de découpage en sous-réseaux associés. Avec la commutation IP et la suppression du service de bout-en-bout ATM, cet empilement correspond à l'intégration d'une signalisation de niveau réseau (3) directement sur la commutation ATM de niveau liaison (2). Il en résulte des éléments de transmission de niveau 2/3 appelés commutateurs/routeurs (*switches/routers*). La commutation IP se place dans un modèle multicouche de "pair-à-pair" (*peer-to-peer*).

Le rapprochement de la commutation et du routage permet de contrôler l'établissement des raccourcis (*shortcuts*). La Commutation IP permet aussi d'éviter la transmission de niveau 3 réalisée dans les routeurs classiques, mais en restant dans l'élément physique associé au routeur pour effectuer cette transmission de niveau 2 localement, commutant directement (*cut-through*). En restant associées à la signalisation IP, les données évitent les problèmes liés à la création de chemins ne respectant pas la topologie du routage. La Figure 3.16 présente l'utilisation d'un *shortcut* avec NHRP et d'un *cut-through* avec la commutation IP.

**CLIP + NHRP :**



**Commutation IP :**



**Fig. 3.16** Comparaison de l'établissement d'un shortcut

Les avantages de la Commutation IP sont les suivants :

- Utilisation de la signalisation IP (pour sa robustesse éprouvée) ;
- Abandon du mode connecté global d'ATM ;
- Gain en performance (au niveau *forwarding*) en utilisant la commutation ATM directement ;
- Simplicité de mise en œuvre (grâce aux couches intermédiaires limitées).

Qui vont de paire avec quelques inconvénients :

- Les propositions sont récentes et on a peu d'expériences ;
- Plusieurs approches concurrentes sont proposées ;
- La gestion du multicast est plus adaptée mais pas complète ;
- Quelques propositions avec de la QoS existent, mais sont statiques.

Parmi les multiples solutions proposées, il ressort deux approches principales : orientée flot ou orientée topologie. Nous classons les solutions IP Switching (Nokia), CSR (Toshiba) et IPSOFACTO (NEC) dans la première catégorie alors que Tag Switching (CISCO), ARIS (IBM) et MPLS (IETF) sont dans la seconde.

Ces nouvelles approches nous intéressent ; nous proposerons au sein du chapitre 5 une solution visant l'intégration du *multicast* et la QoS. Dans la suite nous détaillons d'abord les différentes solutions existantes. Pour une description détaillée des deux approches, voir le livre de Davie [DDR98].

### 3.4.1 Approche orientée flot

Une première catégorie de solutions propose l'utilisation d'une approche basée sur les flots (*flow driven*), c'est-à-dire basée sur l'observation et l'identification de suite de paquets ayant des caractéristiques communes, et ce localement, dans chaque élément intermédiaire.

Le but est de commuter, et non plus de router, les flots reconnus les plus "longs" selon la politique définie. L'élément de base est un routeur associé à un moteur de commutation ATM. On accède au routeur par un VC par défaut. Celui-ci effectue dans un premier temps des fonctions de routage avant de mettre en place la commutation, au niveau des flots identifiés, en dialoguant avec la partie moteur de commutation.

L'intérêt de l'approche *flow driven* repose sur le principe de localité : il n'y a pas de signalisation de bout-en-bout mise en œuvre. Par contre cette approche souffre de sa dépendance à la forme du trafic et du temps de latence avant la commutation. Il en résulte que la proportion de trafic commutée est variable et que l'intérêt de cette approche en dépend largement.

## IP Switching

La solution IP Switching [NLM96, NMLH97, NML97] est soutenue par Nokia (après l'intégration de la startup IPsilicon), et de nombreux industriels.

Cette solution est basée sur des éléments intermédiaires appelés **IP Switches** qui sont composés d'un moteur de commutation ATM (commutateur classique sans la signalisation) et d'un contrôleur gérant toute la signalisation IP. Dans la Figure 3.17, nous représentons une configuration *multicast* avec deux hôtes (A et B) directement connectés au réseau commuté grâce à des cartes ATM et un logiciel adapté. Les membres de réseaux classiques communiquent à travers une passerelle (P). Un **IP Switch** peut donc être vu tel un routeur IP complet avec des interfaces ATM, excepté qu'il héberge en plus des fonctionnalités de commutation.

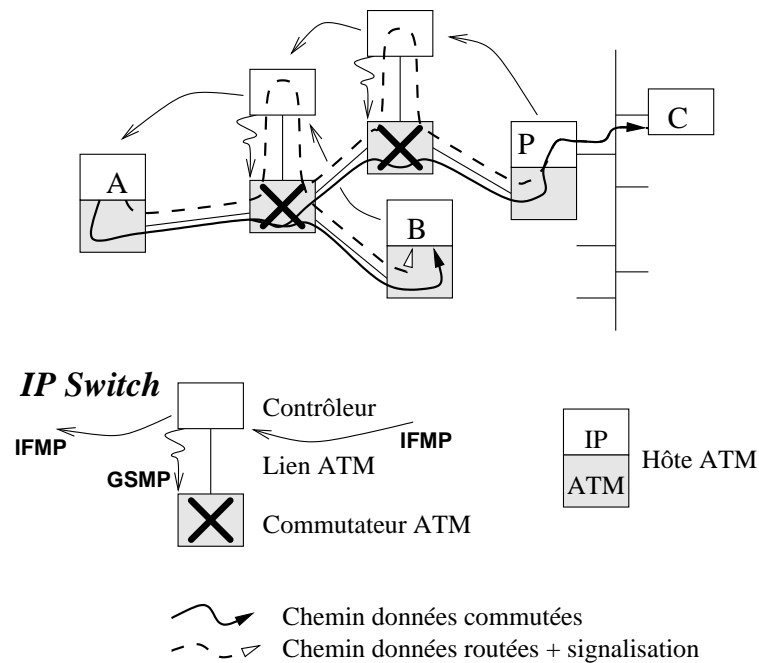


Fig. 3.17 L'architecture d'un IP Switch

Initialement, tous les paquets sont routés à travers le réseau d'**IP Switches** en utilisant des VPI/VCI par défaut pour le trafic IP. Le contrôleur analyse les flots pour les identifier et les qualifier. Les flots intéressants à commuter sont alors associés à un VPI/VCI et cette association est communiquée au nœud précédent. Ce comportement local répété le long du chemin emprunté par le flot va permettre de créer un **chemin commuté** dès lors qu'un flot associé à un VPI/VCI en entrée et sortie d'un **IP Switch** va être directement commuté.

L'approche IP Switching n'utilise donc pas la signalisation ATM et évite ainsi le mode connecté, la résolution d'adresses et la signalisation ATM Forum. Par contre

elle utilise pleinement la signalisation IP standard et deux protocoles supplémentaires :

**IFMP** (*Ipsilon Flow Management Protocol*) Gère la distribution des associations d'identificateurs de flot avec les VPI/VCI correspondants [NEH<sup>+</sup>96a]. Les messages IFMP sont envoyés vers la source de manière à vérifier la bonne transmission de ceux-ci par le nœud précédent (transmission avec le VPI/VCI correspondant).

**GSMP** (*General Switch Management Protocol*) Gère le commutateur ATM (gestion des VPI/VCI et mise en place des *cut-through*). La version d'exploitation est spécifiée dans [NEH<sup>+</sup>96b], et la récente version 2 dans [NEH<sup>+</sup>98]. Les principes sont identiques avec des extensions pour la QoS dans la seconde.

L'encapsulation des flots dans les VC est spécifiée dans [NEH<sup>+</sup>96c]. L'association à des VPI/VCI est temporisée et liée au trafic des flots (*Soft State*). Les paramètres de création et d'activation sont à adapter aux trafics supportés. Les flots sont identifiés selon divers paramètres : application, hôte, voire sous-réseau. Les schémas de la Figure 3.18 détaillent la création d'un chemin commuté. En 1), un premier message IFMP indique le VC à utiliser en entrée. En 2) un second message indique le VC à utiliser en sortie. Et en 3) le *cut-through* est créé.

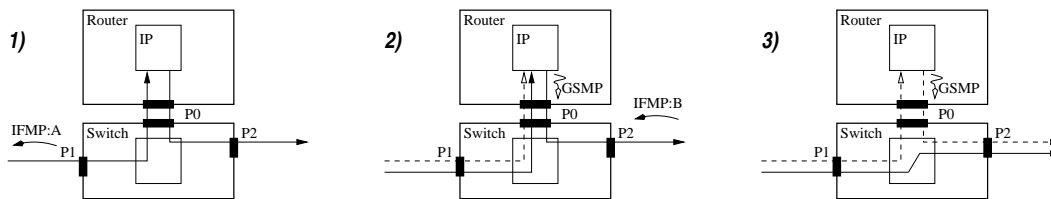


Fig. 3.18 Les étapes de création d'un chemin commuté

Les flots identifiés sont unidirectionnels ainsi que les associations de la table de commutation. Cela permet d'exploiter les capacités multicast intrinsèques. Cette solution est donc naturellement multicast, permettant l'intégration directe du multicast IP mais des problèmes se posent avec l'agrégation (voir la section 3.4.3).

Une simulation de la proportion de paquets commutés a été réalisée à l'aide de traces de trafics dans [LM97]. Cette étude confirme le pourcentage de trafic commuté annoncé (le taux de 80% est généralement atteint). Par contre rien ne nous garantit que les trafics futurs seront aussi bien traités.

### Cell Switch Router (CSR)

La solution CSR [KNE97, EKM<sup>+</sup>96] repose sur un commutateur ATM classique avec la signalisation ATM forum. L'argument de la simplification logicielle n'est plus valable, mais le principe de traitement des flots en mode non connecté est identique

à celui vu précédemment. Dans IP Switching, l'allocation des VC est réalisée de manière locale. La signalisation ATM (Q.2931) est utilisée par le CSR et son interface est UNI. Cela va permettre d'intégrer diverses technologies IP sur ATM entre deux CSR (voir la Figure 3.19).

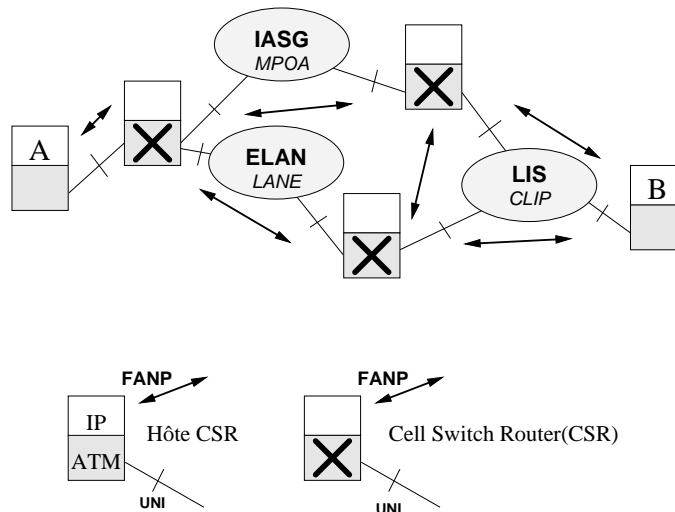


Fig. 3.19 Principes du CSR

Cette approche évite l'utilisation des *shortcuts* NHRP, qui sont remplacés par une séquence de VC connectés au niveau cellule au sein des CSR. Le principe de base est issu d'un des premiers modèles IP sur ATM, appelé COLIP (*COnventional IP over ATM*) [OEN94], qui déconseillait l'utilisation des *shortcuts*.

L'architecture CSR repose sur des routeurs de cellules, qui ont une partie commutateur ATM complète. Une gestion de VC spécifiques pour les *shortcuts* est mise en place. Un protocole de notification appelé FANP (*Flow Attribute Notification Protocol*) [NKS<sup>+</sup>97] est utilisé pour créer les VC, ainsi que la gestion d'un état relâché (*Soft State*) pour la gestion de l'association des VC avec les flots. La création d'un *shortcut* est basée sur le trafic, mais peut aussi être déterminée à l'avance : on se retrouve alors avec un modèle basé sur la topologie que nous verrons dans la section suivante.

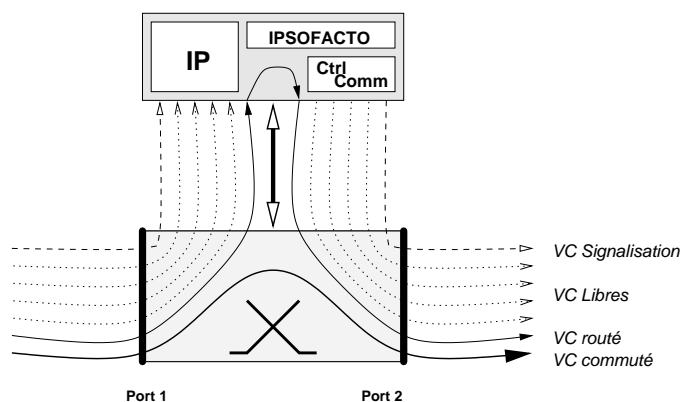
La solution CSR a été réalisée en tenant compte de l'intégration de la QoS, qui est facilitée par l'utilisation native de la signalisation ATM avec ses paramètres de QoS. La gestion de ceux-ci peut être effectuée par une architecture de type RSVP.



## IPSOFACTO

L'approche d'IPSOFACTO [ADA97] diffère des deux précédentes principalement par le fait qu'elle n'utilise pas de protocole pour notifier l'association d'un VC à un flot.

L'architecture est composée d'un contrôleur IP avec un moteur de commutation ATM, de la même manière qu'un IP Switch. Initialement, tous les VC inutilisés sont connectés au contrôleur (voir sur la Figure 3.20) de manière à ce que la détection d'un flot se fasse au moment où des données arrivent sur un VC correspondant à un couple VPI/VCI. Le flot arrivant dans le contrôleur est routé vers le port correspondant et associé en sortie à un VC libre, l'identifiant ainsi avec un nouveau VPI/VCI. Avec toutes ces informations, la commutation peut être effectuée au niveau commutateur tout en conservant la signalisation dans le contrôleur.



**Fig. 3.20** Principes d'IPSOFACTO

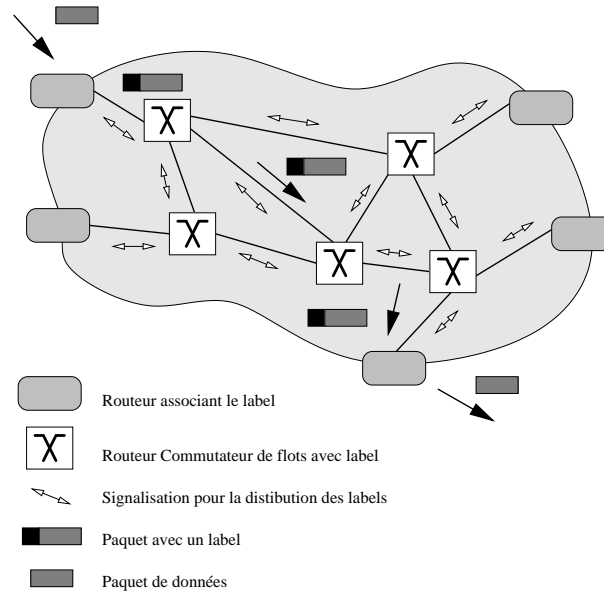
IPSOFACTO est nativement orienté vers le multicast et l'intégration de protocoles tels que PIM, mais il ne gère pas le multiplexage des flots pour des communications N-vers-1.

L'approche suivie par IPSOFACTO nécessite une identification initiale des flots pour l'association avec un VC libre dans le nœud initial.

### 3.4.2 Approche orientée topologie

Une seconde catégorie de solutions utilise une approche basée sur la topologie (*topology driven*), c'est-à-dire basée sur les routes suivies par les données, elles-mêmes issues des informations de routage. L'identification des paquets empruntant un même chemin va permettre de leur attribuer un label de taille fixe évitant la recherche du plus long préfixe (*longest prefix match*) dans la table de routage, de manière à accélérer la transmission au niveau des routeurs. L'association de ces labels est établie

par des routeurs de bordure et la commutation se fait dans la zone avec label par des routeurs/commutateurs (voir la Figure 3.21). Il n'est pas nécessaire que ces derniers puissent gérer le trafic sans label.



**Fig. 3.21** Architecture d'un modèle basé sur la topologie

L'utilisation de cette approche n'est pas uniquement prévue pour ATM. Le label peut être n'importe quel champ situé dans l'en-tête de la couche 2, dans celui de la couche 3, ou dans une couche intermédiaire. Dans le cas de l'utilisation avec ATM, le label est associé au champ VPI/VCI et le gain de performance en transmission sera obtenu grâce à la commutation.

La différence par rapport à l'approche basée sur les flots consiste principalement au fait que l'attribution des labels ne dépend plus du trafic local mais des informations liées à la topologie du réseau. Cela permet de calculer initialement, d'après les tables de routage, les labels qui seront ensuite distribués par une signalisation spécifique, puis utilisés indépendamment de la quantité et des contraintes des paquets les utilisant. L'utilisation des préfixes de routage va permettre de grouper les différents flots allant vers une même destination.

Les intérêts de cette approche résident dans l'absence de temps de latence initial, dans la possibilité d'agréger des flots par préfixe et dans l'indépendance vis-à-vis du trafic (et donc sa stabilité par rapport à un changement de la forme de celui-ci). Par contre, pour le trafic multicast, dont la diffusion vers les groupes virtuels dynamiques n'est pas liée à la topologie, la mise en place de mécanismes basés sur les flots devient nécessaire.

## Tag Switching

L'approche Tag Switching [RDK<sup>+</sup>97, RDS<sup>+</sup>97] repose sur l'association et la distribution de labels liés aux protocoles de routage. Ces labels sont appelés des **tags** et sont associés à un ou plusieurs préfixes de routage. Les tags peuvent être empilés pour refléter la hiérarchie de routage.

La technologie sous-jacente pouvant être variée, le tag peut être inclus dans l'en-tête de la couche 2, de la couche 3 ou dans une couche intermédiaire. Cette solution n'est pas limitée à IP et ATM ; Tag Switching propose l'utilisation de la commutation par label de manière générique.

La distribution des tags le long d'une route est réalisée soit avec un protocole spécifique TDP (*Tag Distribution Protocol*) [DDK<sup>+</sup>97], soit associée à un protocole existant (*piggybacking*) tel que OSPF ou BGP.

La transmission est basée sur la commutation des paquets à partir des tags. Chaque élément du réseau, appelé TSR (*Tag Switch Router*), va contenir une TIB (*Tag Information Base*) dans laquelle le tag d'un paquet entrant est utilisé comme index et correspond à un tag et à un port de sortie.

La Figure 3.22 présente les étapes successives nécessaires avant l'envoi de paquets avec des tags. Le premier schéma indique la distribution des informations de routage, le second la distribution des tags, tandis que dans le dernier, la TIB est complétée par les tags reçus (à chaque préfixe est associé un tag en entrée, le "Lag", un tag en sortie, le "Tag", et une interface de sortie "if").

Pour le multicast, un tag est associé par arbre de transmission. Un protocole tel que PIM peut être utilisé pour la distribution. La section 3.4.3 détaille le multicast utilisé avec la commutation IP.

Un des principaux débouchés du Tag Switching est son utilisation avec la technologie ATM [DDL<sup>+</sup>97]. Les commutateurs ATM sont utilisés directement en tant que TSR, et apparaissent comme de simples routeurs aux éléments de bordure (routeurs associant les tags en périphérie de la zone ATM). Le tag correspond au champ VPI/VCI (ou VCI si on utilise d'autres protocoles) des cellules ATM. L'association de la TIB à la table de commutation ATM permet un transfert efficace. La signalisation utilisée est celle des protocoles de routage avec TDP si nécessaire. La signalisation ATM usuelle, ainsi que l'adressage et le routage ATM Forum ne sont pas utilisés. Dans ce modèle, des limitations apparaissent : la taille du champ VPI/VCI (ou VCI seul) limite le nombre de tags ; la structure à taille fixe de l'en-tête IP ne permet pas l'utilisation de hiérarchie de tags ; et le multicast est délicat à réaliser avec l'utilisation de l'AAL5 qui ne permet pas le multiplexage de plusieurs flots vers un seul.

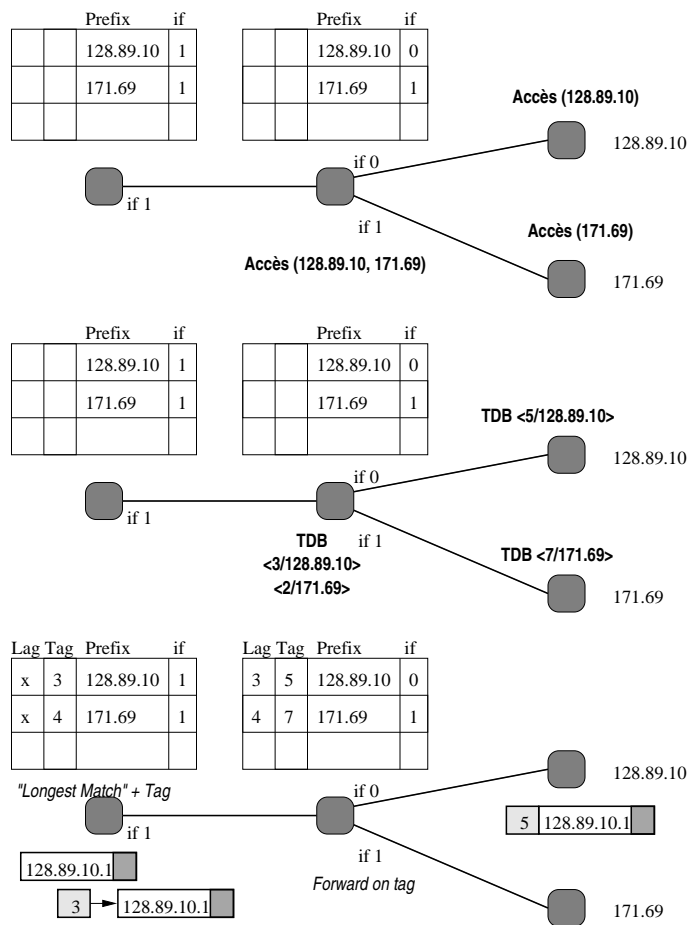


Fig. 3.22 Diffusion des tags et transmission avec Tag Switching

## ARIS

La solution ARIS (*Aggregate Route-based IP Switching*) [VFBW97, FV97] propose l'utilisation de commutateurs ATM avec de la signalisation IP, appelés ISR (*Integrated Switch Router*). Les paquets accèdent à la zone commutée par un ISR entrant (*ingress node*) et sont transmis soit en routage normal, soit en commutation vers un ISR sortant (*egress node*).

Pour établir un chemin commuté, un identificateur unique est diffusé à partir d'un ISR sortant vers les ISR entrants (voir Figure 3.23) et les chemins vers chaque ISR sortant sont créés. Cette approche permet d'agréger le trafic vers un ISR sortant.

## MPLS

Suite aux multiples propositions dans le domaine de la commutation IP, et plus généralement dans celui de la transmission basée sur la commutation par label, un

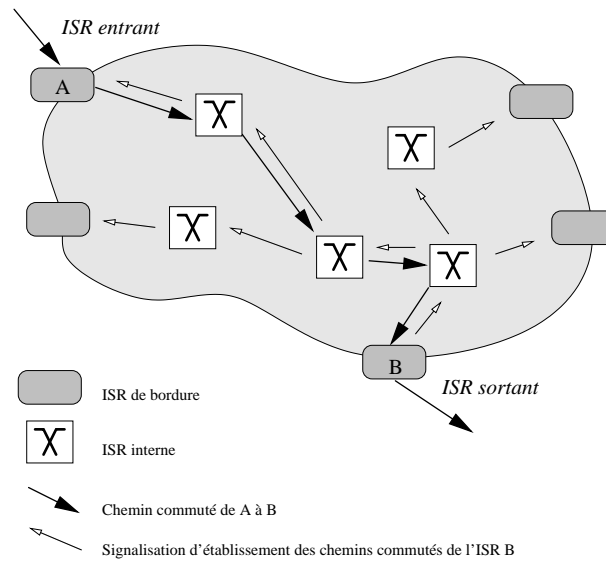


Fig. 3.23 Établissement des circuits commutés avec ARIS

groupe de travail IETF a été créé en mars 1997 sous le nom MPLS (*MultiProtocol Label Switching*). Le but du groupe est de fournir des solutions avec les caractéristiques suivantes :

1. Adaptation au facteur d'échelle du routage de la couche réseau en utilisant de l'agrégation de route.
2. Souplesse des services de transmission avec la possibilité d'identifier certains flots pour des services particuliers (QoS et chemins explicites).
3. Augmentation de la performance grâce à la commutation.
4. Simplification de l'intégration des routeurs et des commutateurs en :
  - (a) réduisant le nombre de voisins d'un élément avec l'équivalence entre routeur et commutateur,
  - (b) rendant les informations sur la topologie physique accessibles à la couche réseau,
  - (c) simplifiant la gestion avec un adressage commun.

La proposition MPLS [CDF<sup>+</sup>97] résulte principalement de la fusion des travaux sur le Tag Switching et sur ARIS, c'est pourquoi de nombreux documents précédents ont été adaptés, tels que LDP (*Label Distribution Protocol*) [ADF<sup>+</sup>98], successeur de TDP.

L'architecture de MPLS [FDFA98] est basée sur la répartition des paquets en FEC (*Forwarding Equivalence Classes*) et leur transmission selon cette classe. Tous les paquets d'une même FEC font partie d'un flux (*stream*) et suivent le même chemin. L'intégration dans un flux se fait dans le nœud d'entrée (*ingress node*) de la zone commutée par l'ajout d'un **label** de petite taille et de longueur fixe. Le label est

utilisé comme index dans chaque LSR *Label Switch Router* pour obtenir le LSR suivant ainsi que le nouveau label. Une signalisation est nécessaire pour distribuer les labels (LDP). Un chemin suivi par un flux de paquets est appelé un LSP (*Label Switched Path*).

Cette solution n'est ni réservée à IP, ni spécifique aux commutateurs ATM. La commutation par label est définie de manière générique, même si actuellement *MultiProtocol* signifie uniquement IP.

L'adaptation de MPLS à la technologie ATM reste l'un des débouchés principaux [DLM<sup>+</sup>97] et souffre des limitations présentées précédemment.

Le multicast reprend l'approche de Tag Switching et est présenté dans la section suivante

### 3.4.3 Comparaison des *multicast*

Pour fournir un service IP multicast, les solutions présentées précédemment mettent en œuvre des mécanismes différents. Nous détaillons le multicast d'IP Switching et celui de MPLS.

La solution **IP Switching** utilise l'approche orientée flot, où le multicast s'intègre naturellement. Chaque flot commence par être routé, évoluant ainsi dans un environnement robuste et maîtrisé. L'association avec un label (VCI/VPI) et la commutation (*cut-through*) se mettent en place progressivement. Les trafics *unicast* et *multicast* sont traités de la même manière et utilisent le même espace d'adressage.

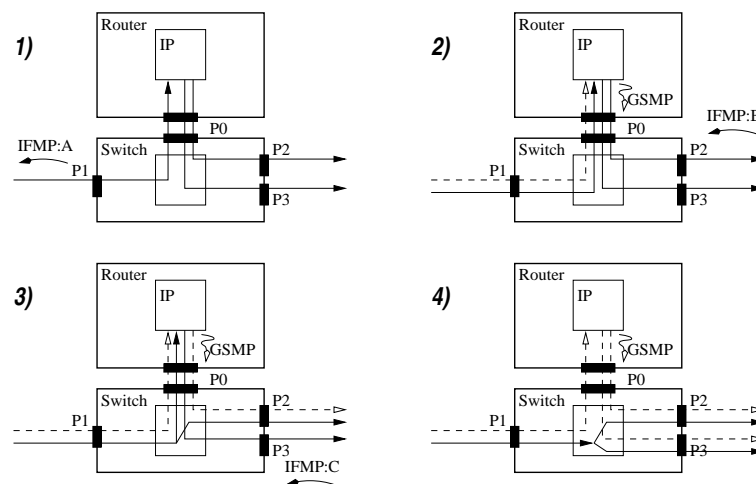


Fig. 3.24 Le multicast avec IP Switching

La Figure 3.24 représente la création d'un *cut-through* multicast selon la technique d'IPsilon. Tous les flots sont considérés comme étant potentiellement multicast et

sont utilisés comme les branches d'un arbre de diffusion potentiel. Les *cut-throughs* sont initiés par ajout de branches, correspondant à des connexions dans la matrice de commutation ATM, du port d'entrée labelé au port de sortie labelé. S'il n'y a pas de connexion déjà présente, une première branche multicast est établie et considérée comme une liaison point-à-point ; si une ou plusieurs branches existent déjà à partir de ce port d'entrée, une branche multicast est ajoutée à cet arbre multicast. La liaison est alors réellement point-à-multipoint. Cet exemple se place dans un environnement IP sur ATM, contexte naturel d'IP Switching. La requête envoyée au moteur de commutation pour l'ajout d'une liaison (ADD BRANCH) commande une action typiquement exécutée dans une matrice gérant le multipoint. Dans la Figure 3.24, les flots sont initialement routés, puis progressivement commutés après avoir reçu leur label. La signalisation reste au niveau du routage et est représentée en pointillés. Les informations sur les labels sont transmises entre les IP Switch par le protocole IFMP. Les communications entre le contrôleur de l'IP Switch et le moteur de commutation de celui-ci sont gérées par le protocole GSMP. L'approche *flow driven* gère les chemins partiellement routés mais ne gère pas l'agrégation de flots.

L'approche MPLS propose avec son approche *control driven* de gérer le trafic multicast séparément du trafic unicast. Suivant MPLS, la table des labels est remplie par un protocole spécifique ou en ajoutant une extension à un protocole existant. La distribution des labels est réalisée par LDP pour les flots unicast. Pour les flots multicast, on va utiliser des arbres recouvrants avec un label spécifique pour chaque groupe. La distribution de labels envisagée est d'utiliser une association avec le protocole de routage multicast PIM.

L'initiation de *cut-through* unicast et multicast est présentée dans la Figure 3.25. La table de routage étendue au label est représentée pour les deux nœuds de gauche. Dans le schéma 1), l'information liée aux préfixes accessibles est distribuée par le protocole de routage. Cette information est intégrée dans les tables avec les interfaces de sortie correspondantes. Dans le schéma 2), LDP ajoute les informations liées au label (Input Label et Output Label) pour le trafic unicast. Dans le schéma 3), les informations obtenues de PIM sont utilisées pour créer une entrée liée à une adresse de groupe, ainsi que le label associé. Le dernier schéma présente la commutation d'un paquet destiné à un groupe multicast. Le paquet est associé initialement au label "6" dans le premier nœud après une recherche du préfixe le plus long (*longest prefix match*) dans la table de routage. Au nœud suivant, ce paquet est commuté selon la table des labels locale : le paquet est envoyé en multicast vers les interfaces de sortie "if 0" et "if 1" avec les labels respectifs "8" et "9".

Avec MPLS, les problèmes liés à l'agrégation pour les communications multipoint-à-multipoint ne sont pas résolus.

MPLS était initialement basé sur l'approche *topology driven* qui lui permettait, grâce à l'agrégation, d'être bien adapté aux environnements de type réseau fédérateur (*backbone*). L'utilisation étendue de MPLS intégrant les environnements locaux est

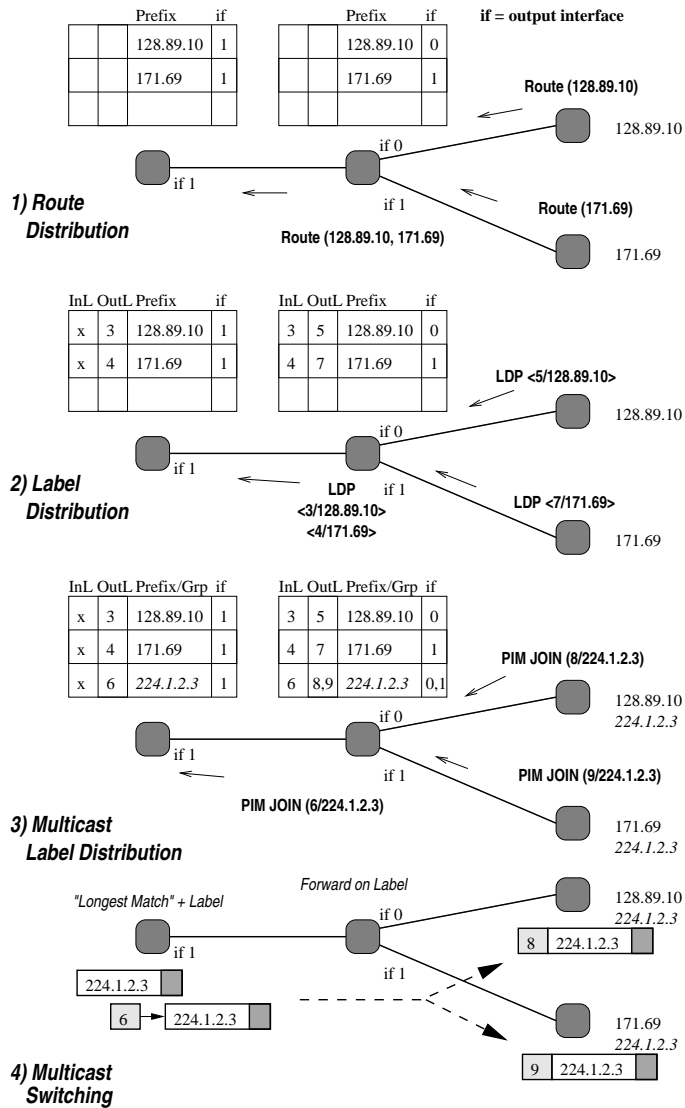


Fig. 3.25 Mise en place du multicast MPLS



d'intérêt moindre car cela va générer des tables de labels de tailles importantes. La séparation des protocoles de contrôle implique, pour utiliser RSVP, la mise en œuvre au sein de MPLS de tous les protocoles qui lui sont relatifs : LDP pour le trafic unicast, PIM pour le trafic multicast et RSVP lui-même.

L'utilisation de l'approche *flow driven* est naturellement dynamique et bien adaptée aux protocoles de gestion de groupes ou de sessions. RSVP a besoin d'une solution adaptée aux flots dynamiques. La solution IP Switching est adaptée mais limitée à un environnement local, tel que les réseaux de campus ou les réseaux privés virtuels (VPN pour *Virtual/Private Network*). Les protocoles existants sont utilisés sans modification pour les trafics unicast et multicast, et RSVP fonctionnera directement sur ces protocoles, comme nous le présenterons dans le chapitre 5.

#### 3.4.4 Perspectives

Le premier but à atteindre est la spécification d'une proposition unique. C'est ce que tend à faire le groupe MPLS, mais seule la tendance orientée topologie semble représentée. Pourtant des réalisations intègrent déjà la combinaison des deux approches, telles que le CSR [KON97] qui permet de fixer à l'avance les routes ou de choisir une allocation dynamique suivant le trafic.

La prochaine évolution notable sera l'introduction de la QoS dans la Commutation IP. Diverses propositions intègrent de la QoS statique attribuée localement. Des travaux visant l'intégration de RSVP et IntServ pour une QoS dynamique seront présentés dans le chapitre 5.

### 3.5 CONCLUSION DU CHAPITRE

Ce chapitre montre que l'intégration d'IP sur ATM est loin d'être triviale. C'est pourquoi les solutions exploitées actuellement (CLIP et LANE) ne proposent que le service IP de base et n'exploitent pas les capacités de la technologie ATM. Les développements récents visent d'une part à intégrer les nouvelles fonctionnalités à travers le *multicast* (MARS) et le haut débit grâce au *shortcuts* (NHRP et MPOA). Les meilleures performances obtenues avec ces *shortcuts* sont mitigées par la complexité qui en découle au niveau routage. De plus, en mode connecté, un multicast efficace s'accorde difficilement de la suppression des routeurs intermédiaires. Pour ces raisons, un nouveau type de solution, la Commutation IP, est apparu. Celle-ci simplifie l'adaptation d'IP sur ATM en utilisant directement la signalisation IP sur la performante commutation ATM.

Pour toutes les approches présentées, l'intégration d'un service multipoint avec QoS est du domaine de la recherche. Nous choisissons de baser nos propositions d'intégration des nouvelles fonctionnalités avec le haut débit sur des solutions présentées dans ce chapitre.

---

Après avoir éliminé l'approche LANE/MPOA pour cause de trop grande complexité, le choix d'un environnement pour développer un service multipoint avec QoS se pose entre l'approche CLIP et la Commutation IP.

- CLIP a l'avantage d'être éprouvé et intégré dans les commutateurs ATM courants, mais il souffre de l'inefficacité de l'adaptation du mode connecté des réseaux ATM. L'ancienneté de cette approche lui procure de nombreux compléments tels le multicast ou IPv6. Nous proposons une solution pour fournir un service multipoint avec QoS efficace dans le chapitre 4.
- La Commutation IP semble attrayante mais elle est récente et les expérimentations débutent à peine. Outre les performances et la simplicité obtenues avec la Commutation IP, une intégration naturelle du multicast et les possibilités de généralisation des approches développées à toutes techniques de commutation justifient la proposition de solutions basées sur cette approche pour le multipoint avec QoS. C'est ce que nous faisons dans le chapitre 5.



## CHAPITRE 4

# Les services multipoints avec “Classical IP”

### Contenu du chapitre

*Ce chapitre présente notre première contribution. Nous nous intéressons à l'intégration du multipoint avec QoS sur les réseaux commutés avec signalisation orientée connexion. Notre proposition intègre l'architecture RSVP sur les réseaux ATM par l'intermédiaire du modèle CLIP. Notre contribution se situe dans l'utilisation de plusieurs arbres multipoints pour permettre la réalisation d'un service multipoint avec QoS hétérogène et renégociable*

*Ce chapitre est organisé de la manière suivante :*

- ⇒ Section 4.1 – Introduction du chapitre – p. 70.
- ⇒ Section 4.2 – L'approche RSVP sur Classical IP over ATM – p. 70 : *Description de l'architecture RSVP pour les réseaux ATM. Présentation des différentes intégrations et des problèmes associés.*
- ⇒ Section 4.3 – Étude de l'intégration du multicast – p. 80 : *Présentation de notre contribution. Définition de la granularité de multicast. Proposition basée sur de multiples arbres multipoints. Mise en œuvre.*
- ⇒ Section 4.4 – Comparaison avec les propositions alternatives – p. 89 : *Présentation de l'approche de l'IETF, du modèle à shortcuts, du modèle à entités de coordination et de la solution MIS pour EARTH.*

- ⇒ Section 4.5 – L’expérimentation – p. 94 : *Présentation de Linux ATM, de la plate-forme MIRIHADE. Description de la réalisation et des expérimentations de notre application de vidéo hiérarchique : LaViD.*
- ⇒ Section 4.6 – Conclusion du chapitre – p. 104.

## 4.1 INTRODUCTION DU CHAPITRE

Le but de ce chapitre est de proposer une solution multipoint avec QoS répondant aux besoins des applications multimédia dans un environnement haut débit. Nous avons choisi comme interface IP, pour son universalité (IP est disponible dans la plupart des systèmes d’exploitation commerciaux), et son extension RSVP pour fournir un service multipoint avec QoS (RSVP sera bientôt intégré dans les principaux systèmes d’exploitation). Le support est basé sur les réseaux ATM, de manière à utiliser leurs services avec QoS en plus des capacités haut débit.

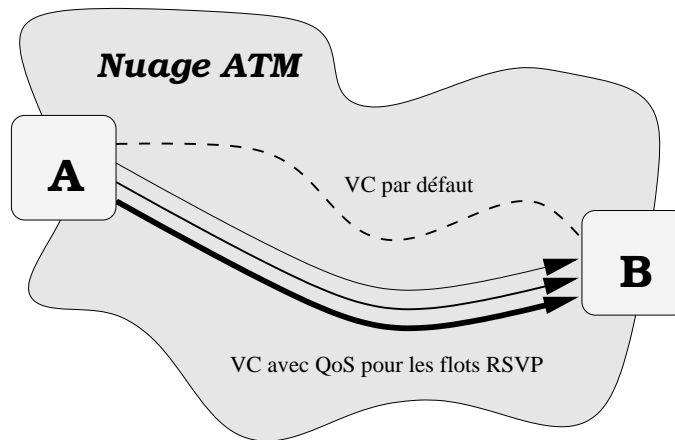
Pour la présente contribution, le choix d’une intégration d’IP sur ATM s’est porté vers la solution promue par l’IETF : *Classical IP over ATM*. Celle-ci s’est imposée à nous comme étant la plus apte à répondre à nos besoins, tant au niveau du choix technologique que de la disponibilité au sein des équipements (voir le chapitre 3). Malgré l’apparition de nouvelles solutions basées sur la commutation IP, ce choix reste judicieux car il s’appuie sur l’une des principales techniques réseaux existantes sur les matériels ATM et est largement soutenu par la communauté de recherche, en particulier dans le groupe de travail ION de l’IETF.

Nous présentons notre solution qui intègre l’ensemble des fonctionnalités de RSVP et qui conserve les caractéristiques majeures associées au service multicast avec QoS d’IP. Celles-ci sont l’hétérogénéité au niveau de la QoS des récepteurs dans une même session et la dynamique avec les aspects de renégociation et de gestion d’un groupe qui évolue dans le temps. Ces fonctionnalités auront pour support indifféremment IPv4 ou IPv6, le service offert étant le même, sur un réseau ATM, et plus particulièrement sur les interfaces disponibles UNI 3.x.

L’approche théorique de cette contribution a été initialement présentée à JDIR’96 [Fou96]. La version étendue de celle-ci, complétée par une présentation des expérimentations effectuées à l’aide une implémentation sur notre plate-forme ATM locale, puis sur la plate-forme nationale MIRIHADE, a été présentée à CFIP’97 [FD97].

## 4.2 L’APPROCHE RSVP SUR *CLASSICAL IP OVER ATM*

Nous nous situons dans un environnement *Classical IP* (CLIP) pour notre intégration de RSVP. CLIP n’assurant que le service IP de base, nous intégrons aussi la proposition MARS (voir la section 3.2.2) pour obtenir le service *IP multicast* sur



**Fig. 4.1** VC dédié à RSVP entre deux éléments connectés à ATM

lequel s'appuie RSVP. Nous présentons dans la suite l'intégration de l'architecture RSVP sur les réseaux ATM, puis les différentes mises en œuvre possibles dans l'environnement CLIP. Nous énumérons ensuite les problèmes liés à ces solutions.

#### 4.2.1 L'architecture RSVP pour les réseaux ATM

L'architecture RSVP est prévue pour des média avec ou sans capacité de QoS. L'intégration habituelle est réalisée avec des réseaux locaux qui sont dépourvus de services avec QoS. Ces services sont alors mis en œuvre à l'aide de mécanismes intégrés aux éléments intermédiaires du réseau. Pour les média proposant des services avec QoS, tels que les réseaux ATM, la gestion de la QoS est transférée au niveau du médium.

Les modèles IP sur ATM ne tiennent pas compte de la QoS (voir le chapitre 3). Entre deux éléments de réseaux connectés par ATM, seule une connexion sans paramètre de QoS est établie. Avec RSVP, il est préférable d'associer les flots à des VC dédiés avec des paramètres de QoS correspondant à la réservation. On évite ainsi la gestion de la QoS des flots sur la connexion initiale, celle-ci étant réservée à la signalisation et au trafic classique. La Figure 4.1 représente l'approche utilisée.

Le modèle RSVP utilisé avec les réseaux ATM est donc semblable à celui utilisé sur les réseaux IP usuels, à ceci près que la gestion des réservations est faite au niveau ATM. Le classificateur de paquets fait correspondre les paquets entrants à leurs VC de sortie au lieu de les affecter à un mécanisme de gestion de la QoS de niveau réseau. L'ordonnanceur de paquets peut être réduit à sa plus simple expression selon la politique de gestion des VC de sortie. Il peut s'occuper du partage des VC selon le style de réservation choisi et assurer le contrôle du respect du contrat de trafic. La structure d'un élément de réseau avec RSVP est représentée dans la Figure 4.2<sup>1</sup>.

<sup>1</sup>Le trafic IP normal et donc la signalisation IP ne sont pas représentés.

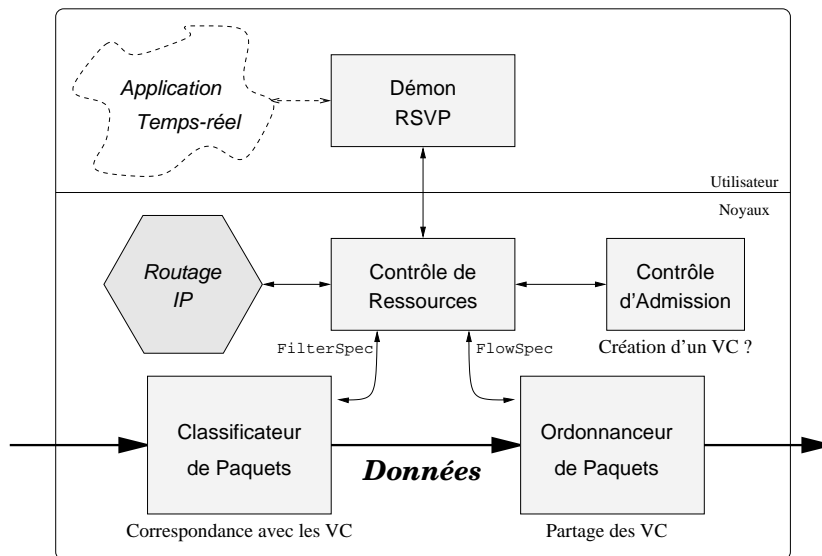


Fig. 4.2 RSVP dans un hôte ou un routeur connecté à ATM

#### 4.2.2 Modèle d'intégration de RSVP sur les réseaux ATM

L'utilisation d'IP sur les réseaux ATM ne va pas sans poser de problèmes, en particulier avec la superposition de deux modèles à connectivité globale où l'on peut établir des connections directes entre tous les éléments connectés, si l'on connaît les adresses des extrémités.

##### Modèle segmenté avec *Classical IP*

Pour la résolution d'adresses dans CLIP, le réseau ATM est segmenté en plusieurs LIS. Dans chaque LIS, les adresses sont résolues avec ATMARP et MARS<sup>2</sup>. Le réseau ATM que nous utilisons peut être une zone intermédiaire au sein d'un réseau IP plus vaste : les éléments en bordure du réseau ATM peuvent être terminaux (hôtes directement rattachés à ATM) ou intermédiaires (éléments d'accès au réseau ATM tels que les routeurs entrants et sortants). Tous les éléments de la partie ATM, qu'ils soient en bordure ou internes, intègrent les composants de l'architecture RSVP.

La configuration de la Figure 4.3 nous sert d'exemple. Elle est composée de deux éléments entrants (A et B) et de deux éléments sortants (C et D). Ces éléments peuvent être terminaux ou non, nous les appelons respectivement émetteurs et récepteurs. La partie ATM représentée est composée de trois LIS reliés entre eux par des routeurs (R1 et R2) gérant RSVP. Les connections potentielles sont représentées par les flèches en pointillé.

<sup>2</sup>Pour la résolution d'adresses multicast avec MARS, nous considérons qu'un *cluster* correspond à un LIS.

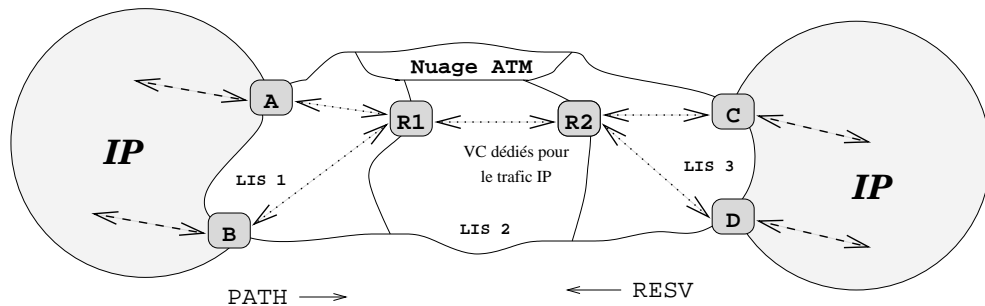


Fig. 4.3 Modèle de base

L'intérêt de ce modèle est de représenter simplement une topologie multipoint-à-multipoint pour l'étude des réservations selon leurs styles et la demande des récepteurs.

### Modèle à connectivité ATM globale

Le modèle précédent n'est pas optimal en terme de connectivité ATM entre les éléments entrants et sortants. Des connections de bout-en-bout (*shortcuts*) au niveau de la zone ATM sont possibles en utilisant NHRP. L'utilisation de RSVP est transparente sur ce modèle. La gestion de la QoS dans la partie ATM est plus efficace mais l'utilisation de ces liaisons directes n'implique pas l'optimalité en terme de ressources utilisées, car il n'y a plus d'agrégation dans la zone ATM. Les connexions multipoints ATM étant limitées au modèle 1-vers-N, les agrégations ont seulement lieu au niveau des flots arrivant dans le routeur entrant et sont transmises aux N routeurs sortants en un saut au niveau IP à travers ATM.

La Figure 4.4 représente deux éléments entrants (A et B) et deux éléments sortants (C et D). Tous les membres attachés au réseau ATM sont directement accessibles. Des *shortcuts* sont réalisés aussi bien pour le trafic IP classique que pour les flots RSVP.

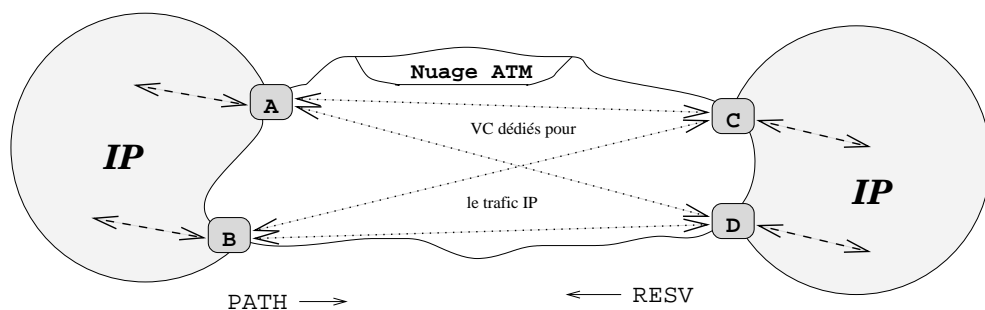


Fig. 4.4 Modèle à connectivité globale



L'utilisation de *shortcut* directement au niveau IP rend la gestion de *shortcuts* spécifiques à RSVP (voir la solution proposée dans la section 4.4.2) inutile car RSVP utilise des liaisons directes de manière cohérente avec IP. Cette approche pose plutôt des problèmes au niveau de la topologie de routage qu'elle entraîne.

Pour notre solution, nous nous plaçons dans le modèle CLIP avec LIS sans *shortcuts*, ce modèle étant facilement simplifiable à un modèle à connectivité ATM globale.

### Modèle serveur avec *Classical IP*

Les fonctionnalités de RSVP sont de niveau 3. Pour les intégrer dans un réseau ATM considéré de niveau 2, il faut des entités qui assurent la remontée protocolaire. Dans le modèle RSVP sur ATM avec des LIS, les fonctionnalités RSVP sont assurées par les routeurs inter-LIS qui mettent en œuvre les mécanismes RSVP au niveau IP. Dans le modèle serveur, des entités indépendantes sont utilisées pour gérer les fonctions RSVP.

Nous nous intéressons à un modèle avec Serveurs RSVP (S.R.) indépendants (voir la solution proposée dans la section 4.4.3). Les S.R. sont jumelés avec des MARS pour résoudre le multicast. Il peut y avoir plusieurs S.R. dans un LIS. Les éléments de bordure du nuage ATM sont configurés pour accéder à un S.R. proche. Ils envoient les requêtes RSVP vers ce serveur qui va gérer la création des VC correspondant aux réservations.

Sur la Figure 4.5 avec ses éléments entrants (A et B), sortants (C et D) et ses deux routeurs (R1 et R2), nous avons intégré un S.R. par LIS. Les connexions avec les S.R. représentent les liens pour la signalisation. Les messages PATH et RESV vont passer sur ce réseau dédié.

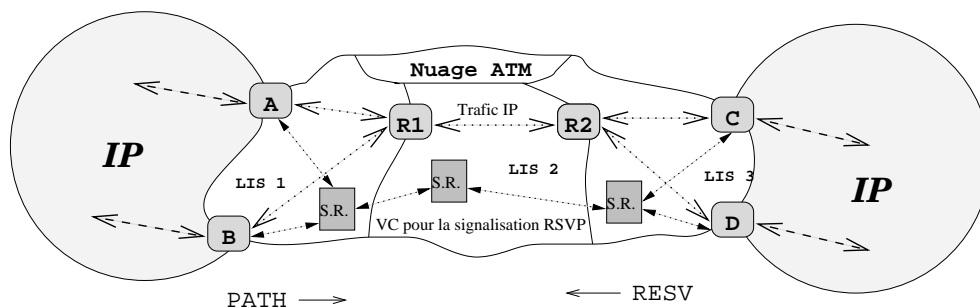


Fig. 4.5 Modèle serveur avec LIS

Différents mécanismes sont à mettre en œuvre pour gérer le serveur et ses communications avec les autres serveurs : la réplication (sécurité), la détection, la reconfiguration, l'élection et la diffusion de l'adresse du serveur, le partage de charge, l'accès

direct aux serveurs des autres LIS sans passer par un routeur intermédiaire (dans ce modèle on dissocie complètement le flux IP de celui des serveurs RSVP).

Ce modèle nécessite la mise en place d'un protocole assez lourd qui n'est peut être pas justifié dans un modèle avec LIS où les fonctionnalités RSVP peuvent être facilement intégrées dans les routeurs.

### Modèle serveur à connectivité ATM globale

Le modèle serveur, en cas de connectivité globale, permet d'intégrer des éléments intermédiaires pour obtenir les fonctionnalités RSVP dans le nuage ATM, et permettre ainsi l'utilisation de l'agrégation dans la zone ATM.

Sur la Figure 4.6, toujours avec ses éléments entrants (A et B) et sortants (C et D), les membres attachés au réseau ATM global sont directement accessibles. Nous introduisons une structure de serveurs RSVP (S.R.) composée ici de deux serveurs. Les liens sur les S.R. sont utilisés pour la signalisation RSVP séparée du flot IP normal.

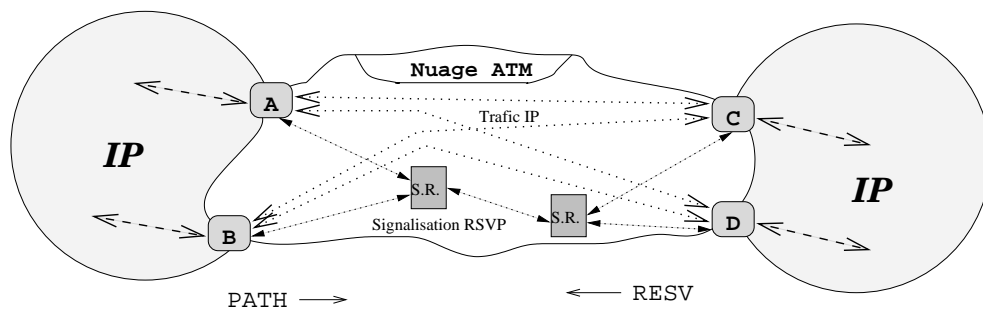


Fig. 4.6 Modèle serveur en connectivité globale

Nous pouvons faire, sur la mise en œuvre, les mêmes remarques que précédemment. Par contre, la nécessité d'utiliser plusieurs serveurs en connectivité globale est justifiée car on n'a plus d'éléments intermédiaires comme dans le modèle précédent. Une structure distribuée avec plusieurs serveurs pourra, tout en évitant les problèmes de congestion, intégrer les fonctionnalités RSVP rendues impossibles dans le nuage ATM, telles que l'agrégation.

### Réservations de VC ATM selon les styles RSVP

Après avoir détaillé le modèle de création des VC, nous présentons le partage des VC selon les différents types d'agrégation de réservations (dans notre modèle RSVP sur ATM, les réservations correspondent à des VC).

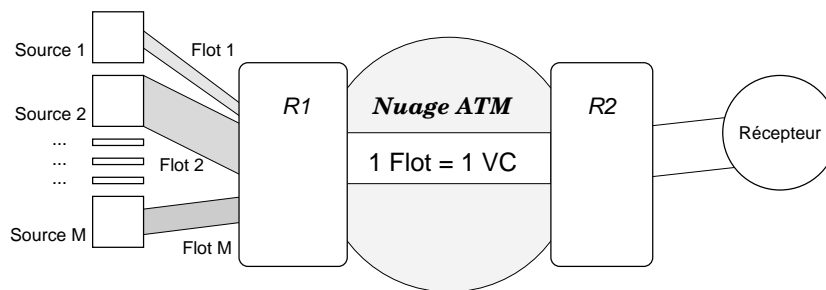
Les styles indiquent le type d'agrégation possible entre les flots d'une session<sup>3</sup>. Les filtres<sup>4</sup> sont utilisés pour attribuer les paquets arrivant dans un élément RSVP à leurs réservations respectives. Nous énumérons dans la suite l'association des flots et des VC au sein d'une session en fonction du style de la réservation :

**Style *Wildcard-Filter*** Une réservation est partagée par un groupe d'émetteurs (la réservation correspond à la demande la plus importante).

*Représentation :*

**WF(\*Q)**

*Astérisque pour l'émetteur quelconque et Q pour le flowspec demandé.*



**Fig. 4.7** VC pour le style Wildcard-Filter

Dans la Figure 4.7, nous représentons une session avec M émetteurs et un style de réservation WF. L'agrégation des flots produit un flot avec pour réservation le "maximum" des réservations demandées par le récepteur. Au niveau ATM cela se traduit par un VC avec des paramètres de QoS correspondant à la réservation agrégée. Le changement du maximum (nouvelles réservations, arrivées ou départs de récepteurs ou d'émetteurs) peut amener à renégocier ce VC.

**Style *Shared-Explicit*** Une réservation est partagée par un groupe d'émetteurs et les émetteurs sont déclarés explicitement (la réservation correspond à la demande la plus importante).

*Représentation :*

**SE((S<sub>1</sub>, S<sub>2</sub>, ...)Q)**

*S<sub>n</sub> les émetteurs nommés et Q pour le flowspec demandé.*

Dans la Figure 4.8, nous représentons une session avec M émetteurs et un style de réservation SE. L'agrégation des flots se produit entre les N flots (N ≤ M) explicitement sélectionnés dans le filtre, produisant un flot ayant pour réservation le "maximum" des réservations demandées par le récepteur par rapport aux N émetteurs. Les autres flots sont agrégés dans un flot avec une QoS non spécifiée. Au niveau ATM cela se traduit par deux VC : un VC avec la QoS "maximum" demandée par le

<sup>3</sup>Session = (Adresse destination IP, Protocole, Port destination)

<sup>4</sup>Filtre = (Adresse source IP, Port source)

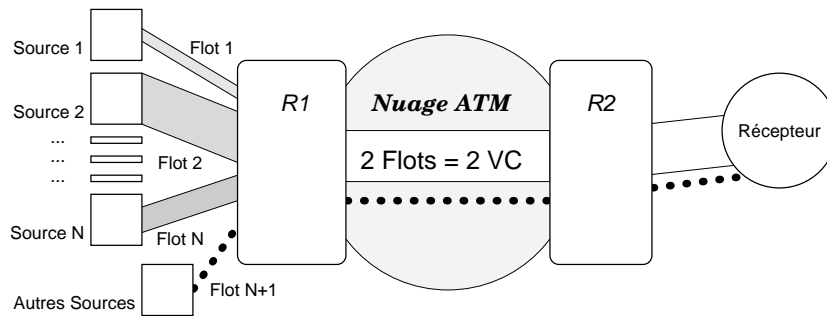


Fig. 4.8 VC pour le style Shared-Explicit

récepteur par rapport aux  $N$  émetteurs choisis et un flot sans QoS pour les paquets des autres émetteurs (en pointillés sur la figure). Le changement du maximum des  $N$  émetteurs peut amener à renégocier le VC avec QoS.

**Style *Fixed-Filter*** Une réservation distincte par émetteur (pas de partage).

*Représentation :*  $\mathbf{FF}(S_1 Q_1), \mathbf{FF}(S_2 Q_2), \dots$

$S_n$  les émetteurs nommés et  $Q_n$  pour le flowspec correspondant.

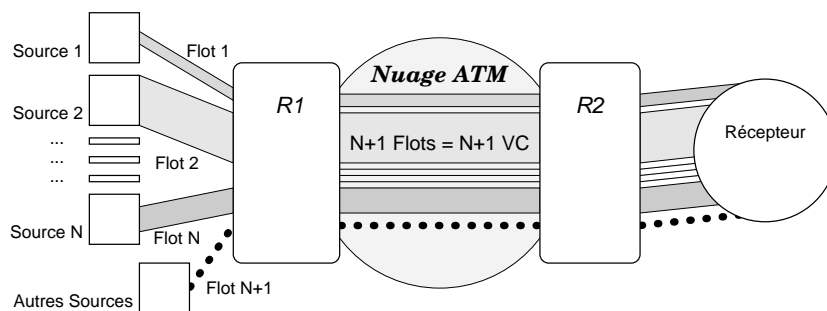


Fig. 4.9 VC pour le style Fixed-Filter

Dans la Figure 4.9, nous représentons une session avec  $M$  émetteurs et un style de réservation FF. Il n'y a pas d'agrégation entre les  $N$  flots ( $N \leq M$ ) explicitement sélectionnés dans le filtre, ce qui conduit à  $N$  réservations distinctes. Les autres flots sont agrégés dans un flot à la QoS non spécifiée. Au niveau ATM cela se traduit par  $N+1$  VC :  $N$  VC avec les QoS demandées par le récepteur respectivement pour  $N$  émetteurs choisis et un flot sans QoS pour les paquets des autres émetteurs. Les changements parmi les  $N$  émetteurs peuvent amener à ajouter, détruire ou renégocier des VC avec QoS.

### 4.2.3 Les problèmes liés à l'adaptation de RSVP sur ATM

Les approches orthogonales utilisées par les réseaux IP et ATM posent de nombreux problèmes à résoudre pour obtenir une solution d'adaptation efficace. De même la multiplicité des intégrations d'IP sur ATM laisse présager de nombreuses difficultés et des choix architecturaux à effectuer.

#### Shortcuts ATM

La question de savoir quand utiliser les *shortcuts* reste ouverte. Faut-il favoriser l'agrégation de flots ou la création de *shortcuts* de taille maximum pour avoir les meilleures performances ? Plus on utilise de *shortcuts*, moins on remonte au niveau IP ; mais on ne peut plus gérer les agrégations de flots à cause de l'absence de cette fonctionnalité au niveau ATM.

L'utilisation de *shortcuts* au niveau RSVP impose la modification du protocole : les messages PATH et RESV doivent transporter des informations pour la création de ces VC dont ils ne pourront pas résoudre l'adresse à l'autre extrémité. Une gestion adaptée dans les nœuds intermédiaires est aussi nécessaire pour que les nœuds évités par le *shortcut* ne modifient pas les messages qui continuent à transiter par le chemin IP. La prise en compte des *shortcuts* au niveau RSVP n'est donc pas transparente.

La gestion des *shortcuts* est ici associée à RSVP, or c'est un problème de routage. On peut donc s'interroger sur l'orthogonalité du protocole de réservation et du protocole de routage et se demander si l'on traite bien les *shortcuts* au bon niveau sachant que dans le modèle initial de RSVP, la gestion des flots est en fonction du routage et non l'inverse. L'utilisation des *shortcuts* dans RSVP a été mise en place à ce niveau pour pallier un manque des protocoles de routages.

#### Élément Intermédiaire

Plus une solution a d'éléments RSVP intermédiaires, plus son potentiel d'agrèger des flots augmente. Une solution intégrant des éléments intermédiaires permet d'agrèger au mieux les flots RSVP. Cette fonction est réalisée au niveau 3 à cause du filtrage, pour identifier les flots. Mais pour utiliser pleinement l'agrégation, il faut étendre l'interface ATM pour inclure deux nouvelles fonctionnalités :

- Avec des réservations partagées, il faut prendre la réservation maximum. Un nouveau récepteur qui rejoint le groupe avec une demande supérieure nécessite la possibilité de changement dynamique de QoS du VC utilisé.
- La gestion de récepteurs hétérogènes implique l'utilisation d'une QoS adaptée par branche. Il faut donc un nouveau modèle de VC multicast où l'on pourra spécifier la QoS de chaque branche que l'on crée.

Avec l'agrégation des flots, la gestion du modèle multipoint-à-multipoint devient possible, mais la gestion n'est pas transparente pour l'ATM et la mise en œuvre est lourde. Ici encore, on associe à RSVP des fonctionnalités qui devraient se trouver dans les couches inférieures.

### **Adaptation du Contrôle de Trafic IntServ à la QoS ATM**

Les spécifications du trafic ou de la QoS ne sont pas intégrées à RSVP qui ne tient pas compte du contenant des réservations qu'il opère. De nombreux travaux sur la conversion de ces valeurs IntServ en paramètres de QoS ATM ont été réalisés (voir en particulier [BFGK95, Gué96, GB98]).

### **Adaptation du *Soft State* RSVP au *Hard State* ATM**

Les réseaux ATM gèrent un état global du système (*Hard State*). RSVP, par contre, met en œuvre un état global relâché (*Soft State*) pour pallier le manque d'état global dans les réseaux IP. L'adaptation de RSVP peut en tenir compte ; par exemple, la gestion des envois périodiques de messages pour le maintien du *Soft State* peut ne plus être utile dans le nuage ATM.

### **UNI 3.x ou 4.0**

L'interface UNI 3.x est actuellement largement disponible sur les divers matériels ATM. L'interface UNI 4.0 est une évolution dont l'avancée nous concernant le plus est le LIJ. Cette fonctionnalité ne remet pas en cause les mécanismes que l'on peut mettre en œuvre sous UNI 3.x. En effet, le LIJ évite de faire remonter une demande de création de VC jusqu'à la racine de l'arbre de diffusion, mais au niveau de la signalisation, les requêtes remontent l'arbre multicast pour diffuser les messages de réservation en amont.

### **Limitations du Modèle Multicast ATM**

La plupart de solutions proposées utilisent directement le multicast ATM sans analyser les problèmes liés à ses limitations. Trois problèmes majeurs subsistent :

**Modèle Multicast ATM** Pour identifier une session multicast avec ATM, on utilise la racine de distribution, car le modèle est orienté émetteur. Les sessions multicast ATM sont point-à-multipoint. RSVP suppose un modèle sous-jacent orienté récepteur qui va lui permettre de gérer des sessions multipoint-à-multipoint. Le modèle utilisé par RSVP est celui d'IP, avec abstraction de groupes : on découple les adresses des groupes des récepteurs eux-mêmes pour en faire des entités virtuelles dynamiques, sur lesquelles n'importe quel émetteur peut envoyer des informations et

auxquelles les récepteurs peuvent appartenir à tout moment. Les sessions multicast RSVP sont identifiées par les groupes de récepteurs.

**Dynamacité** Le modèle de réservation de RSVP est dynamique : il permet la renégociation à tout moment des réservations associées aux flots d'une session. Avec ATM, les flots correspondent à des VC et les réservations aux QoS des VC. La dynamacité du modèle est basée sur celle de la QoS des VC, or la QoS d'un VC ATM est statique. La gestion de la renégociation d'un VC est donc basée sur le remplacement du VC avec les paramètres de QoS obsolètes par un autre VC avec de nouveaux paramètres. Deux solutions sont envisageables pour cela :

- ouverture d'un nouveau VC et fermeture de l'ancien (coût de la renégociation basé sur la création d'un VC),
- changement de VC dans le cas où différents VC avec différentes QoS sont initialement créés (coût de la renégociation basé sur le changement de VC).

**QoS Hétérogène** Le modèle RSVP est orienté récepteurs et permet à chacun de ceux-ci de demander une réservation adaptée à leurs besoins, de manière indépendante les uns des autres. La construction d'arbres de distribution avec des branches aux QoS différentes est donc implicite. Or, au niveau ATM, les interfaces UNI 3.x et 4.0 spécifient uniquement la création de VC point-à-multipoint avec des QoS Homogènes. La création d'un VC point-à-multipoint est séquentielle : création d'une branche initiale vers un premier récepteur avec une QoS donnée, puis ajout successif des autres branches vers les autres récepteurs avec la même QoS.

### 4.3 ÉTUDE DE L'INTÉGRATION DU MULTICAST

Suite aux remarques présentées dans la section précédente, nous choisissons une approche simple basée sur la vision initiale de CLIP. Nous n'intégrons pas les *shortcuts* de manière à conserver les routeurs intermédiaires nécessaires à un multicast efficace. Nous n'utilisons pas non plus d'éléments supplémentaires pour éviter le découplage entre la signalisation RSVP et la signalisation IP. L'efficacité du multipoint ne doit pas provenir d'une augmentation du nombre d'éléments intermédiaires mais d'un découpage judicieux du nuage ATM en sous-réseaux. Au niveau du contrôle de trafic, nous proposons d'utiliser les développements du groupe ISSLL [GB98].

Dans ce contexte, le problème à résoudre est de trouver un modèle multipoint avec QoS adapté à RSVP et permettant de pallier les limitations intrinsèques du modèle multipoint des réseaux ATM. Nous présentons ainsi dans cette section l'étude théorique de notre proposition. Celle-ci est basée sur de multiples arbres multipoints pour fournir un service multicast avec QoS dans un environnement CLIP (avec l'extension MARS).

### 4.3.1 Granularité de service multicast

Dans le modèle RSVP, les réservations doivent permettre des connexions multipoint-à-multipoint avec des récepteurs aux réservations variées. L'adaptation de ce modèle à ATM pose le problème des arbres multicast à QoS variées, sachant que les VC point-à-multipoint d'ATM sont à QoS homogène (identique sur toutes les branches). Nous définissons la **granularité de service multicast** comme la capacité à proposer une QoS proche de celle demandée sur chaque branche. Nous étudions ici trois types de granularité à partir de notre modèle de base (Figure 4.3). Pour chaque cas, nous présentons une description du modèle puis nous analysons successivement l'hétérogénéité et la dynamique des récepteurs avant de discuter des avantages et des inconvénients.

#### Granularité faible – Réservations homogènes

La solution la plus simple est l'utilisation directe du service point-à-multipoint d'ATM. L'utilisation de réservations homogènes implique que nous nous basions sur la QoS "maximum" de manière à satisfaire tous les récepteurs.

**Hétérogénéité** Nous utilisons le modèle point-à-multipoint d'ATM qui est homogène en QoS. Il faut mettre en place une politique pour évaluer la QoS globale allouée à toutes les branches.

**Dynamisme** En cas de changement de QoS, il faut fermer et recréer l'ensemble du VC point-à-multipoint, après avoir calculé une nouvelle QoS. Suivant la politique mise en œuvre on peut laisser le VC inchangé, si la branche concernée est toujours sur-dimensionnée. Par contre, l'ajout et la suppression d'un récepteur acceptant les paramètres de QoS choisis pour le VC existant est dynamique (pour l'ajout il faut cependant compter avec le délai de création d'un VC).

**Exemple** Nous avons représenté les VC ainsi créés sur la Figure 4.10 : il n'y a que des VC point-à-multipoint (*simplex*), même s'il n'y a qu'un nœud suivant (en prévision de l'ajout d'un nouveau successeur). Nous supposons qu'il y a une agrégation de type WF entre R1 et R2.

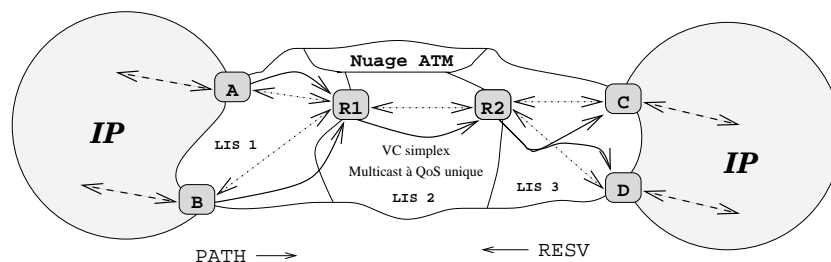


Fig. 4.10 Réservation avec une QoS unique



**Avantages** Une granularité de service multicast faible correspond à une grande simplicité et à une signalisation légère grâce à l'utilisation du service point-à-multipoint fourni par l'interface actuelle d'ATM.

**Inconvénients** La mise en place d'une politique pour trouver la QoS globale est difficile. Le modèle RSVP étant par nature dynamique, un élément intermédiaire reçoit des messages RESV à n'importe quel moment. Doit-il attendre un certain nombre de messages avant de réserver, doit-il se baser sur le premier message RESV, avec le risque d'initialiser un VC sous-dimensionné ? Toutes les branches de l'arbre multicast ne vont peut-être pas accepter un VC avec cette QoS "maximum". Le résultat de la négociation à l'aide des messages PATH et RESV nous fournit la QoS à choisir : avec les messages PATH, chaque récepteur obtient la QoS potentiellement disponible auprès des émetteurs, majorée par celle que le chemin peut lui fournir. En retour le nœud concerné devrait allouer la QoS "maximum" demandée dans toutes les branches du VC multicast majorée par la plus petite QoS disponible sur ces branches, *or c'est une information dont il ne dispose pas*. Il est donc nécessaire d'allouer un VC point-à-multipoint avec la QoS "minimum" pour s'assurer de la création complète de celui-ci. Ce modèle de réservation par maximum a les mêmes inconvénients que l'agrégation du style WF : il faut renégocier tout le VC si le maximum de QoS change. Le choix de cette solution dépend du nombre de récepteurs, de leur variété et de leur dynamique. Nous pouvons aussi choisir de sur-dimensionner le VC pour parer à une augmentation de la QoS demandée, nous avons alors un modèle à allocation statique avec une sur-allocation qui risque d'être très importante avec un grand nombre de récepteurs.

Ce modèle de réservation, souvent envisagé dans les solutions RSVP sur ATM, ne satisfait ni la dynamique ni l'hétérogénéité nécessaires au modèle RSVP.

### **Granularité fine – Réservations par VC point-à-point**

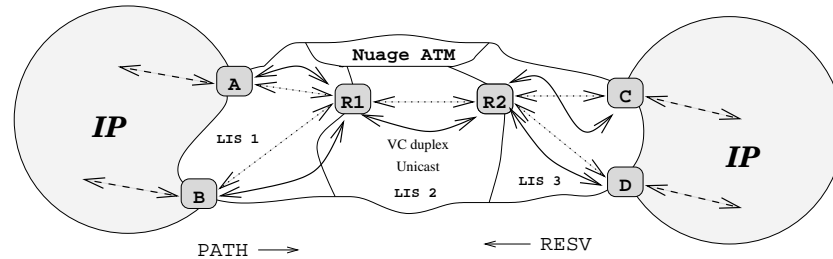
Le modèle de réservation opposé au précédent est celui des réservations totalement hétérogènes : pour s'adapter au plus près à chaque réservation, on réalise au niveau ATM des VC point-à-point avec la QoS correspondante pour chaque successeur du nœud concerné. Pour ce modèle, le passage des messages PATH et RESV nous donne toutes les informations pour allouer les VC.

**Hétérogénéité** Nous n'utilisons plus les VC point-à-multipoint ATM, mais des VC point-à-point pour obtenir la meilleure granularité : la QoS est négociée sur la base d'une QoS pour un récepteur, chaque branche ayant la QoS désirée si celle-ci est disponible.

**Dynamisme** Nous utilisons des VC point-à-point ATM, qui ont une QoS fixée. Lors d'un changement de QoS, il nous faut fermer l'ancien VC et en ouvrir un autre avec la QoS demandée. Le changement est plus léger que précédemment, mais on est encore loin de la dynamique du modèle RSVP qui est basée sur des

changements de réservations internes aux routeurs et non sur des réservations de circuits.

**Exemple** Dans la Figure 4.11, nous avons représenté les VC point-à-point (*duplex*) avec la QoS désirée, entre les nœuds successifs de l'exemple. Nous supposons qu'il y a encore une agrégation de type WF entre R1 et R2.



**Fig. 4.11** Réservation avec QoS multiples

**Avantages** L'intérêt de ce modèle, en plus d'être complètement hétérogène, est qu'en cas de renégociation, nous ne recréons que le VC point-à-point de la réservation qui a évolué.

**Inconvénients** Une granularité de service multicast fine correspond à une signalisation élevée et une inadaptabilité au facteur d'échelle des groupes multicast. Un autre défaut de ce modèle est la sur-utilisation de ressources qu'entraîne la création de  $N$  connexions point-à-point par rapport à une connexion point-à-multipoint (mais cette ressource sera adaptée au mieux aux besoins de la session). De plus, nous définissons ici des VC bidirectionnels alors que les réservations RSVP sont unidirectionnelles (basées sur des flots *simplex*)<sup>5</sup>.

Ce modèle de réservation, même s'il est hétérogène, ne répond pas à la dynamique nécessaire au modèle RSVP. De plus, la non-utilisation des capacités multipoints du niveau ATM empêche l'adaptation à des groupes multicast avec un nombre de récepteurs élevé.

### Granularité intermédiaire – Réservations par multiples VC point-à-multipoint

Les deux modèles de réservation précédents représentent les deux manières extrêmes d'adapter des réservations hétérogènes sur ATM. En observant qu'en général le nombre de QoS demandées est limité et que nous pouvons les classer en paliers, nous définissons un modèle mixte. Celui-ci combine plusieurs arbres multicast ATM homogènes avec des QoS correspondant à ces paliers.

<sup>5</sup>Précisons cependant que les VC ATM point-à-point sont asymétriques, et peuvent donc avoir un débit réservé nul pour le sens "retour".

Pour chaque niveau de QoS choisi, nous allons donc définir un arbre multicast, matérialisé au niveau ATM par un VC point-à-multipoint avec les paramètres de QoS correspondants. Un récepteur change dynamiquement d'arbres multicast selon la QoS dont il a besoin. Nous envisageons deux modèles que nous décrivons plus précisément dans la section suivante : le **modèle à flots séparés** où le récepteur utilise une branche sur l'arbre dans lequel se trouve la QoS qu'il demande (les récepteurs rejoignent l'arbre multicast avec la QoS recherchée), et le **modèle à codage hiérarchique** où le récepteur utilise une branche de **tous** les arbres dont il a besoin (les récepteurs rejoignent les N premiers arbres multicast pour réaliser le niveau de QoS voulu).

**Hétérogénéité** La possibilité d'avoir plusieurs QoS est assurée par les différents niveaux de QoS des arbres multicast (et complémentaire si le codage est hiérarchique).

**Dynamisme** La dynamique est assurée par la facilité de changement d'arbres multicast. Les arbres étant créés initialement, un récepteur peut demander son rattachement aux arbres qui correspondent à la QoS qu'il demande.

**Exemple** Dans la Figure 4.12, nous avons représenté deux arbres multicast avec deux QoS différentes initiées par A et B vers les récepteurs. Les deux flots arrivent au niveau de C et D, à partir desquels ils peuvent être sélectionnés par les éléments en aval. Nous supposons qu'il y a une agrégation de type WF entre R1 et R2.

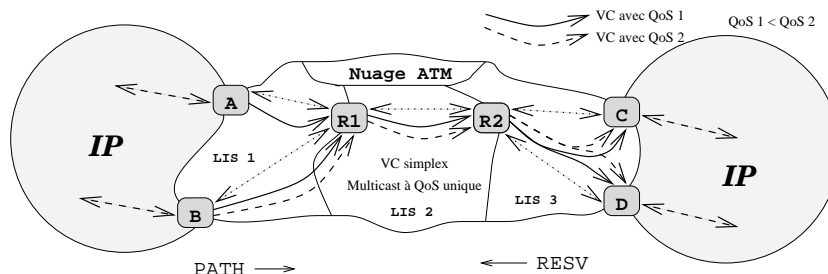


Fig. 4.12 Réserve avec plusieurs VC point-à-multipoint

**Avantages** La dynamique et l'hétérogénéité se rapprochent de celles recherchées pour RSVP. Les mécanismes multipoint du niveau ATM sont conservés. En gérant les rattachements aux arbres multicast à l'aide de gestion de groupe comme nous l'expliquons en section 4.3.2, nous restons transparents par rapport à RSVP, de manière à diminuer les modifications nécessaires.

**Inconvénients** Ce modèle reposant sur le principe de paliers de QoS, il faut définir le nombre et les caractéristiques de ces paliers. Pour faire des agrégations, il faut que les niveaux de QoS choisis correspondent entre les différents émetteurs. La gestion de plusieurs arbres multicast dans le nuage ATM peut entraîner une surconsommation des ressources.

Ce modèle pallie les deux problèmes que nous avons identifiés précédemment (section 4.2.3). L'utilisation d'un VC ATM point-à-multipoint pour chaque niveau de QoS permet d'obtenir l'hétérogénéité. La dynamicité est alors assurée par le changement d'arbres.

#### 4.3.2 Mise en œuvre du modèle à arbres multiples

Dans cette section nous expliquons les choix de base de notre modèle, en étudiant plus précisément deux variantes et en expliquant la mise en œuvre. Nous proposons ensuite une modification du modèle privilégiant la dynamicité.

##### Choix préalables

Nous considérons que la séparation d'un flot en plusieurs sous-flots correspondant à des QoS différentes est du ressort de l'application, qui est la seule à connaître la sémantique du flot de données. Nous ne nous préoccupons pas de la manière de générer ces flots, notre modèle reste ainsi très générique. De plus, en cas de ressources limitées du réseau, il reste possible de n'émettre que les flots correspondant aux QoS les plus faibles.

Nous choisissons d'associer une adresse de groupe IP à chaque niveau de QoS. Ceci nous permet de nous reposer sur les mécanismes de routage multicast pour la distribution des différents flots. Ainsi les arbres multicast pourront être ajustés au mieux suivant la répartition des récepteurs, la traversée de zones non-RSVP pourra se faire de manière transparente, et nous soulageons RSVP des fonctionnalités de filtrage de flots. Enfin IP multicast a l'avantage d'être opérationnel et largement utilisé.

##### Multiple VC point-à-multipoint avec flots séparés

L'application génère autant de flots que de niveaux de QoS désirés. Chaque flot a une QoS différente et est indépendant. Les différents flots sont routés à travers le réseau de façon à ce que les destinataires reçoivent le flot correspondant à la QoS demandée. Sur la Figure 4.13, l'émetteur A envoie un flot FA1 avec la QoS 1 et l'émetteur B, qui génère un flot avec une meilleure QoS, envoie deux flots FB1 et FB2, avec respectivement la QoS 1 et la QoS 2 (le flot FB1 transporte les mêmes données que FB2 mais avec la QoS 1, plus faible que la QoS 2). Ces flots sont agrégés dans le routeur R1 (nous supposons une agrégation de type WF) et sont transmis dans deux VC point-à-multipoint assurant les QoS 1 et 2 vers R2. C reçoit les deux flots, D seulement la QoS 1 (on suppose qu'après D personne ne demande la QoS 2). Les applications peuvent récupérer au niveau de C l'une ou l'autre des QoS en adhérant au groupe de FA1 et FB1 pour la QoS 1 ou au groupe de FB2 pour la QoS 2.

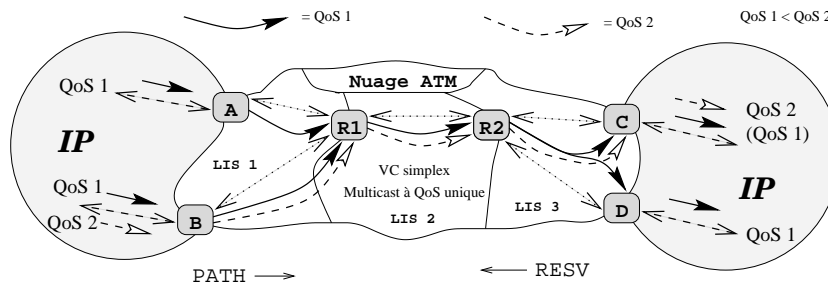


Fig. 4.13 Réserve de VC point-à-multipoint avec flots séparés

Un problème est posé par la séparation des flots. Prenons le cas d'un récepteur qui choisit de recevoir la QoS 2. Si un émetteur émet à la QoS 1 plus faible, le récepteur ne recevra plus rien. Une solution serait de propager le flot de QoS 1 dans le canal de la QoS 2 à ce moment là, mais cela signifie faire de l'agrégation de flots **inter-session**, ce qui est contraire au modèle RSVP (les flots de notre modèle appartiennent à des sessions différentes au sens de RSVP). Si nous voulons résoudre ce problème en restant conforme sur ce point au modèle RSVP, les récepteurs doivent demander à recevoir aussi le flot de niveau inférieur à la QoS qu'ils demandent (sur la Figure 4.13 c'est le cas du récepteur alimenté par C).

### Multiple VC point-à-multipoint avec codage hiérarchique

L'application génère plusieurs flots construits selon le principe du codage hiérarchique. Les différents flots sont routés à travers le réseau vers les destinataires qui en ont besoin. Pour atteindre les récepteurs, nous assignons une adresse de groupe multicast par flots. Cela permet au protocole de routage multicast d'ajuster les arbres multicast au mieux selon la répartition des récepteurs à la sortie du nuage ATM. Pour une QoS donnée, un récepteur peut avoir besoin de N flots. A la réception, l'application doit reconstruire le flot avec la QoS demandée selon le principe du codage hiérarchique utilisé, avec les N flots obtenus en s'abonnant aux N groupes correspondants. Au niveau de l'agrégation, le procédé est transparent pour RSVP étant donné qu'il a lieu sur chaque flot de manière indépendante. Sur la Figure 4.14, l'émetteur B génère un flot avec une QoS 3 maximum qu'il hiérarchise en 3 flots FB1, FB2 et FB3 (en utilisant les trois flots, la QoS 3 est obtenue; avec FB1 et FB2, la QoS 2; et avec FB1 seulement, la QoS 1). B envoie ces trois flots vers R1. L'émetteur A envoie un flot FA1 avec la QoS 1. Les deux flots avec la même QoS, FA1 et FB1, appartenant à la même session, sont agrégés dans le routeur R1 dans un VC de QoS 1 et sont transmis vers R2 (nous supposons ici une agrégation de type WF). Ce flot ainsi que FB2 et FB3 sont transmis vers R2 sur trois VC point-à-multipoint. Dans notre exemple (Figure 4.14), une application connectée à C demande les flots de niveau 1 et 2, une connectée à D ne demande que le flot de niveau 1, enfin une

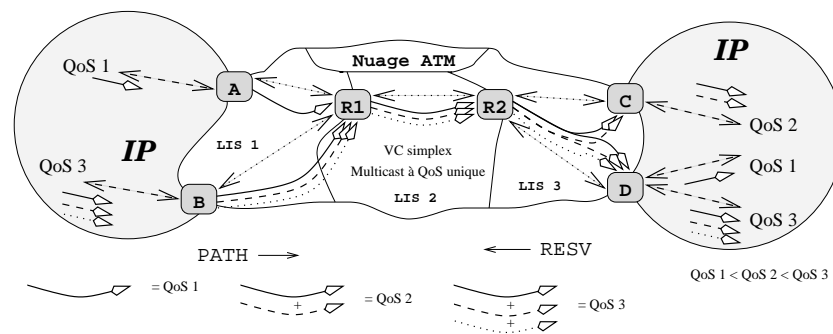


Fig. 4.14 Réserve de VC point-à-multipoint avec codage hiérarchique

dernière connectée à D demande les trois niveaux. C peut transmettre les flots de niveau 1 et 2, et D tous les niveaux grâce aux VC originaires de R2.

### Utilisation des arbres multiples

Pour recevoir un flot avec une QoS donnée, un récepteur s'abonne au groupe correspondant ce qui lui permet de recevoir les messages PATH; il renvoie alors un message RESV pour effectuer sa demande de réservation. Si le récepteur a besoin de plusieurs flots (cas du codage hiérarchique) il devra s'abonner à chacun des groupes correspondants, il recevra alors les messages PATH pour chaque flot et pourra effectuer ses réservations en renvoyant des messages RESV. Les messages PATH et RESV sont gérés de manière normale à l'extérieur du nuage ATM comme à l'intérieur, les messages utilisant les VC réservés à IP (non fiables). Quand un récepteur quitte un groupe, il ne recevra plus de message PATH et ne renverra donc pas de message RESV. Le noeud RSVP qui ne recevra plus de message RESV de ce récepteur supprimera alors la branche correspondante du VC point-à-multipoint.

Les récepteurs sélectionnent le niveau de QoS désiré en adhérant ou en quittant les groupes multicast correspondants. Si de nouveaux récepteurs se joignent ou quittent la session, cela modifie le routage dans le nuage ATM, les tables de MARS changent et les messages PATH sont diffusés vers le nouveau groupe de récepteurs en même temps que l'on crée ou détruit des VC point-à-multipoint.

Il ne faut pas fermer les VC à cause d'un arrêt de réservation, la ressource devant rester disponible pour une autre réservation. On ne peut pas non plus fermer les VC sur demande explicite des récepteurs pour être capable de s'adapter aux groupes multicast. Il faut mettre en œuvre une gestion basée sur les messages PATH. Les arbres sont créés par rapport aux groupes auxquels ont adhéré les récepteurs et ceux-ci sont répercutés dans la base de MARS. C'est par rapport à cette table que l'on sait s'il y a encore des membres concernés par un arbre multicast.

Si l'on peut conserver la transparence vis-à-vis de RSVP pour les arbres multicast à branches hétérogènes, il n'en va pas de même pour la dynamique. Le modèle

conventionnel RSVP sur ATM associe les VC aux réservations et donc crée les VC au moment de la réservation. Si nous supposons que les VC point-à-multipoint sont créés avant la demande de réservation, cela permet les changements de VC sans attente de création de circuits. Cette modification par rapport à la gestion habituelle RSVP sur ATM est étudiée dans la section suivante.

Au niveau de l'application, une politique d'envoi des messages PATH correspondant au plus petit flot (de qualité minimum) en premier peut être mise en œuvre pour assurer tout d'abord le passage d'un minimum (un VC de taille supérieure peut être rejeté); les autres flots viennent ensuite si l'on peut créer les autres VC.

### Dynamacité et pré-réservation

Quand un récepteur demande à passer à un niveau de QoS supérieur, l'ajout d'une branche au VC point-à-multipoint qui transporte le flot correspondant peut être nécessaire; sauf dans le cas particulier où le récepteur est hors du nuage ATM et est connecté à ATM par un routeur sortant qui alimente déjà un autre récepteur avec ce flot. Or le temps d'établissement d'un VC ATM n'est pas garanti, et il peut être très variable. Cette constatation nous amène à proposer une variante de notre modèle, utilisant une **pré-réservation au niveau ATM**. A la réservation, les récepteurs spécifient le niveau de QoS qu'ils demandent, plus un ensemble de niveaux de QoS qu'ils souhaitent recevoir plus tard, les niveaux de QoS potentiels. Ils reçoivent alors le flot de QoS demandé, et à l'intérieur du nuage ATM les arbres multicast sont construits pour tous les niveaux de QoS potentiels déclarés. Les flots correspondant aux niveaux de QoS potentiels arrivent jusqu'en bordure du nuage ATM, où ils sont abandonnés (les routeurs sortants doivent gérer l'abandon d'un flot) .

Les récepteurs se voient ainsi assurés d'une dynamacité maximum pour les niveaux de QoS potentiels, au détriment d'une sur-réservation relative de ressources. Ceci peut également être utile dans le cas où un récepteur est capable d'anticiper son besoin de QoS supérieur. Ainsi, l'adhésion au groupe correspondant aura pour effet de mettre en place la branche manquante du VC ATM. Au moment précis où le changement de QoS devient nécessaire, le message RESV pourra établir la réservation dans la dernière partie du chemin et le flot sera disponible.

Au niveau de la mise en œuvre, ceci impose que les VC ATM soient gérés par les messages PATH. On aura donc un comportement différent à l'extérieur du nuage ATM (création des réservations par les messages RESV) et à l'intérieur du nuage ATM (création des réservations par les messages PATH) où le modèle sera orienté émetteur.

## 4.4 COMPARAISON AVEC LES PROPOSITIONS ALTERNATIVES

Quelques autres propositions abordent aussi l'adaptation des services RSVP sur ATM. Nous les comparons avec notre proposition, en insistant sur leur manière de traiter les réservations RSVP et plus particulièrement du point-de-vue du multi-points :

- La première proposition est soutenue par l'IETF au sein de son groupe de travail ISSLL et repose sur une approche standard de l'intégration RSVP où les réservations sont effectuées entre chaque élément RSVP adjacent [CBB<sup>+</sup>98],
- La seconde proposition utilise la connectivité globale au sein du nuage ATM [BFGK95],
- La troisième proposition présentée est basée sur l'utilisation d'éléments intermédiaires gérant les fonctionnalités RSVP au niveau d'un cluster MARS [OS96],
- Une dernière proposition est basée sur l'utilisation de la proposition EARTH au lieu de MARS et utilise des serveurs MIS combinant les fonctionnalités liées au multicast, à RSVP et à la QoS [SCS<sup>+</sup>97].

### 4.4.1 Approche RSVP sur ATM de l'IETF

L'IETF s'intéresse aux problèmes liés au transport du trafic IP avec contraintes temporelles sur ATM depuis quelques années [BCDB95]. Avec l'introduction de RSVP et des IIS, le groupe de travail IISLL (*Internet Integrated Services over Specific Layer*) a été créé pour les adapter aux diverses technologies de réseau. En particulier une proposition a été développée pour ATM.

L'approche proposée se base sur les techniques de transport du trafic IP *Best Effort* sur ATM définies dans [CSV96] : on y trouve par exemple CLIP avec une intégration similaire à celle que nous avons choisie. La signalisation passe avec le trafic IP. Les routeurs intègrent RSVP, les fonctions de gestion des VC et la traduction des paramètres de QoS de IntServ vers ATM. Dans la Figure 4.15, nous nous positionnons sur le modèle CLIP vu précédemment. Les messages PATH vont de l'émetteur (A) vers le récepteur (B) en construisant le *Soft State* dans les éléments intermédiaires. Les messages RESV parcourent le chemin inverse. A la réception de ces messages, des VC correspondant aux réservations sont créés.

Cette méthode se base sur l'utilisation classique de RSVP avec un médium gérant la QoS. L'adaptation aux différentes approches IP sur ATM est plus ou moins évidente selon que l'on intègre aussi les propositions avec le *multicast* ou les *shortcuts*. Quatre documents sont actuellement disponibles :

**A Framework for Integrated Services and RSVP over ATM** C'est le document de base, qui introduit les nombreuses possibilités et problèmes associés à



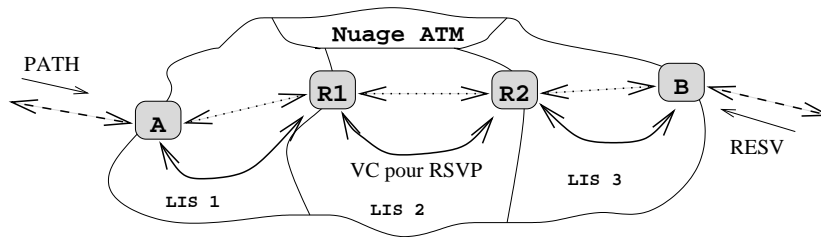


Fig. 4.15 Réserveation classique à travers trois LIS

l'approche classique sans apporter de solutions particulières. Les problèmes tels que l'utilisation des *shortcuts*, de la renégociation de QoS ou des connexion multipoint-à-point sont cités [CBB<sup>+</sup>98].

### Interoperation of Controlled-Load and Guaranteed Service with ATM

Ce document présente de manière détaillée les divers services et paramètres de QoS ATM à travers UNI 3.x et UNI 4.0, tout en proposant leurs adaptations aux classes de services de l'IntServ [GB98].

**RSVP over ATM Implementation Guidelines** Cette note précise la mise en œuvre RSVP sur un service ATM à base de connexions dynamiques (SVC) [Ber98a].

**RSVP over ATM Implementation Requirements** Cette note est associée à la précédente et précise les services requis pour assurer la compatibilité entre les différentes solutions RSVP sur des SVC ATM [Ber98b].

L'approche de l'IETF est une extension naturelle de RSVP sur les réseaux ATM. Elle est la base des travaux visant cette intégration. Elle diffère de notre approche par sa limitation aux capacités de l'ATM, particulièrement pour le multipoint.

#### 4.4.2 Modèle avec liaisons ATM directes

Le modèle proposé dans [BFGK95] se positionne sur CLIP mais contrairement au modèle précédent, celui-ci utilise des *shortcuts* au niveau RSVP. Pour utiliser la QoS de bout en bout, la remontée au niveau IP dans les routeurs intermédiaires est évitée. Les routeurs qui créent les *shortcuts* doivent connaître les adresses ATM d'entrée et de sortie du nuage pour créer la connexion directe. Lorsqu'un message PATH arrive dans un routeur entrant (A), celui-ci transmet en plus l'adresse ATM de ce routeur. Les routeurs intermédiaires dans le nuage ATM ne vont ni créer d'état ni modifier le champ nœud précédent. Lorsque le routeur de sortie reçoit le PATH il crée un état et y ajoute l'adresse ATM contenue dans le PATH. Il est alors possible de créer un *shortcut*, soit en renvoyant un message RESV au routeur entrant avec l'adresse ATM du routeur sortant par le chemin classique, pour lui indiquer de créer un VC entre A et B (Figure 4.16) avec la QoS demandée, soit en créant le VC entre B et A directement par le routeur sortant (B). Les VC étant bidirectionnels, la réservation

de ressources se fait dans les deux sens ; mais la réservation sera spécifiée seulement pour A vers B.

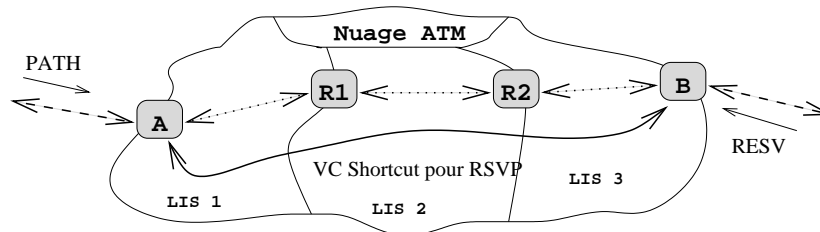


Fig. 4.16 Réservation d'un "shortcut" à travers trois LIS.

L'utilisation de *shortcut* permet une utilisation maximum des capacités ATM. Les nœuds R1 et R2 sont évités sauf pour la signalisation. Deux inconvénients apparaissent ici :

- La gestion des *shortcuts* est un problème de routage, or les spécifications de RSVP insistent sur le fait que RSVP ne gère pas le routage. C'est au niveau IP de gérer les *shortcuts*.
- Ce que l'on gagne au niveau ATM avec les *shortcuts* est perdu au niveau optimisation de l'utilisation des ressources : moins il y a d'éléments RSVP intermédiaires, moins l'agrégation de flots est possible.

### Intégration de RSVP sur ATM pour le Multicast

Le multicast avec les *shortcuts* pose problème. La restitution du service multipoint-à-multipoint de RSVP dans ATM passe par l'utilisation des fonctionnalités multipoints de celui-ci. La solution proposée est basée sur des *shortcuts* point-à-multipoint. Deux approches sont proposées selon que l'on utilise les fonctionnalités RIJ (UNI3.x) ou LIJ (UNI4.0).

**Shortcuts créés par l'émetteur (RIJ)** On procède comme précédemment en intégrant la résolution des adresses multicast. Le message PATH arrive aux routeurs sortants B et C avec l'adresse ATM de A. Supposons que B reçoive le message en premier. L'adresse de B se trouvera dans les messages RESV à destination de A. Les routeurs intermédiaires R1 et R2 ne modifient pas le message. L'arrivée de ce message au routeur entrant A permet l'établissement d'un VC point-à-multipoint ayant pour racine A et pour feuille B. Le second message RESV provenant de C arrive en A. A va ajouter au VC existant une nouvelle feuille, C, grâce au message de signalisation ATM ADD PARTY. La Figure 4.17 nous indique l'état final de la réservation.

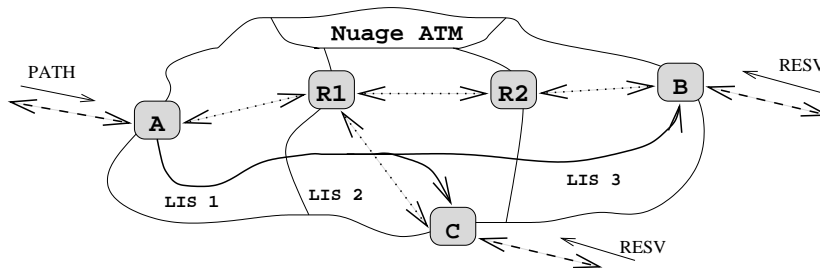


Fig. 4.17 Réserve d'un "shortcut" Multicast à travers 3 LIS.

**Shortcuts créés par les récepteurs (LIJ)** Cette seconde approche spécifique à UNI 4.0 utilise la possibilité qu'a un récepteur ATM de rejoindre un VC point-à-multipoint s'il connaît l'adresse de la racine et l'identificateur de l'arbre multicast. Les messages PATH sont échangés comme précédemment et continuent d'utiliser les mécanismes IP multicast sur ATM. Ces messages vont transporter, en plus de l'adresse ATM de A, un identificateur de l'arbre auquel les récepteurs devront se joindre avec la requête LIJ. Lorsqu'un routeur sortant génère un message RESV, il peut être le premier à le faire. B, par exemple, envoie une requête LIJ. Le VC est créé avec A comme racine et B comme première feuille. Les routeurs sortants suivants, tels que C, vont rejoindre l'arbre en rajoutant des branches à l'arbre multicast avec des requêtes LIJ. La racine A n'en sera pas avertie évitant ainsi la congestion à la racine.

Cette solution permet d'utiliser au mieux les possibilités multipoints des réseaux ATM, mais elle s'éloigne du modèle multicast de RSVP en évitant les routeurs intermédiaires. Elle est évidemment limitée au modèle multipoint d'ATM, c'est-à-dire arbre point-à-multipoint avec des branches aux QoS homogènes et statiques (si un récepteur demande une QoS plus importante, il faut renégocier tout l'arbre au niveau de tout le nuage).

#### 4.4.3 Entités de coordination RSVP

Une autre approche a été développée pour mieux utiliser le multicast avec RSVP : un modèle avec des entités intermédiaires [OS96]. Le principe de base de ce modèle est d'intégrer des éléments de coordination RSVP dans le nuage ATM. Les fonctions réseau sont supportées par ces RCE (*RSVP Coordination Entity*), éléments qui s'interfacent avec ATM via UNI et gèrent les fonctionnalités RSVP telles que :

- agrégation de flots RSVP (un ou plusieurs RCE sur un chemin),
- établissement et gestion d'arbres *multicast* au niveau ATM,
- support de l'hétérogénéité des récepteurs,
- conversion des Tspec en paramètres de trafic ATM,
- conversion des Rspec en qualité de service ATM.

MARS gère la résolution des adresses IP de groupe. Il peut y avoir un ou plusieurs RCE dans un cluster MARS. La Figure 4.18 représente une session RSVP avec deux émetteurs (S1 et S2), un RCE (RCE1), trois récepteurs (R1, R2, R3). Ce modèle est facilement extensible à plusieurs RCE. Les émetteurs et les récepteurs peuvent ne pas être des éléments terminaux et dans ce cas ils font suivre les messages PATH et RESV.

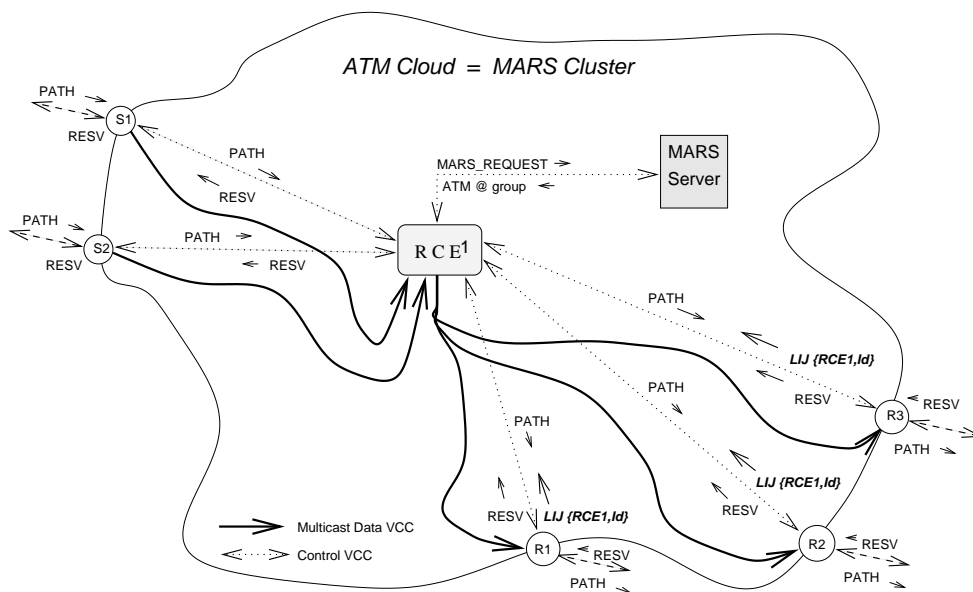


Fig. 4.18 Gestion d'un flot par les RCE.

Les fonctionnalités de filtrage et d'agrégation de flots sont réalisées au niveau du RCE. Ce modèle est plus efficace au niveau des fonctionnalités supportées, mais de nombreux éléments intermédiaires doivent être traversés, limitant ainsi les performances. De plus ce modèle doit faire face à la complexité de mise en place d'une approche basée sur des serveurs. Même si l'introduction de serveurs spécifiques à RSVP permet une meilleure gestion du multipoint avec QoS, l'utilisation de connexions point-à-multipoint uniques limite ce modèle au service des réseaux ATM.

#### 4.4.4 MIS et EARTH

MIS (*Multicast Integration Server*) [SCS<sup>+</sup>97] est une solution basée sur une alternative de MARS : la proposition EARTH (voir la section 3.2.2) basée sur un serveur de résolution d'adresses multicast unique au sein d'un nuage ATM. À ce serveur EARTH, un serveur RSVP est associé pour la gestion de la QoS au niveau 3 (le serveur EARTH gère déjà la QoS de niveau 2). L'ensemble de ces deux serveurs

forme le MIS. Tous les participants du MLIS<sup>6</sup> ouvrent un VC de contrôle vers le MIS pour le trafic EARTH et celui lié à RSVP.

La séquence opérationnelle de la proposition MIS est la suivante : l'émetteur envoie un message PATH au MIS qui va le diffuser aux récepteurs ; les récepteurs intéressés renvoient un message RESV au MIS, qui le convertit en signalisation EARTH (niveau 2), pour que l'émetteur puisse créer directement un VC point-à-multipoint avec QoS entre lui et les récepteurs. Une fois le VC créé, un acquittement (niveau 2) est renvoyé au MIS qui le convertit en acquittement de niveau 3 pour l'émetteur. En résumé, un échange classique de messages RSVP a lieu au niveau 3 à travers le MIS, mais au passage du message RESV dans le MIS, celui-ci génère un échange de messages du niveau 2 pour la création du *shortcut* ATM.

Cette solution promeut un modèle qui découple non seulement le trafic RSVP, mais aussi ceux du multicast du trafic IP géré par CLIP. L'intégration de RSVP sur ATM est du type serveur et utilise des *shortcuts* gérés spécifiquement par RSVP. On retrouve les limitations habituelles liées au service multipoint des réseaux ATM (pas d'agrégation ni de récepteurs hétérogènes) avec en plus le risque de débordement de l'unique serveur.

## 4.5 L'EXPÉRIMENTATION

Nous procédons à la validation de notre approche théorique par une mise en œuvre expérimentale. Au sein du LIP6, nous disposons d'une plate-forme ATM locale et avons participé, au moment des expérimentations, à la plate-forme ATM nationale MIRIHADÉ. Notre travail s'inscrit donc dans les nombreux tests effectués sur ce réseau ATM. Les machines utilisées dans ces environnements sont des PC sous GNU/Linux avec des cartes d'interface ATM.

Dans la suite nous décrivons les moyens disponibles et mis en place pour la réalisation de notre expérimentation. Nous présentons en particulier une application qui repose sur notre proposition de multiples arbres multipoints.

### 4.5.1 Système GNU/Linux ATM

Le choix de machines s'est porté sur des stations de travail économiques et performantes pour l'utilisation en réseau : des PC Pentium avec des cartes ATM supportent notre expérimentation<sup>7</sup>. Le système d'exploitation utilisé est le système GNU/Linux avec une extension pour le support d'ATM.

---

<sup>6</sup>LIS dédié à tous les participants aux communications multicast du nuage ATM.

<sup>7</sup>Le facteur coût est important pour expérimenter des solutions multipoints à cause du nombre de machines nécessaires.

Type de VC	Classe de services	Signalisation associée
<b>PVC point-à-point</b>	CBR et UBR	—
<b>SVC point-à-point</b>	CBR et UBR	UNI 3.0 et UNI 3.1

**Tab. 4.1** Les types de VC supportés par le *driver* de l'EPFL

## Le système d'exploitation GNU/Linux

Le système d'exploitation GNU/Linux est de type Unix. Il est supporté par une très large communauté de développeurs. La totalité des éléments associés au système GNU/Linux est disponible gratuitement avec les sources sous licence de type GPL (*GNU General Public Licence*)<sup>8</sup>.

D'autres systèmes d'exploitation proposent un environnement libre avec sources similaire à GNU/Linux (avec les sources gratuites), tels que NetBSD ou FreeBSD, mais ils ne disposent pas du support d'ATM à l'époque de l'expérimentation.

### *Driver* Linux ATM

Depuis 1995, l'EPFL (École Polytechnique Fédérale de Lausanne) développe le support d'une plate-forme ATM [Alm96] sur GNU/Linux pour les communautés de la recherche et de l'enseignement. Des gestionnaires de cartes ATM et des implémentations de protocoles ont été réalisés et sont disponibles au sein de la distribution Linux-ATM.

Les cartes ATM que nous utilisons font parties du matériel supporté. Ce sont des cartes **Efficient Networks ENI155** pour port PCI. Ces cartes supportent des débits de 155 Mbps sur fibre optique multimode.

Les types de connexions supportés sont présentés dans le Tableau 4.1.

Les éléments de signalisation suivants ne sont pas supportés :

- Les classes de service VBR et ABR pour tout type de connexion,
- Les **connexions point-à-multipoint**,
- La signalisation UNI 4.0.

Les services et protocoles décrits dans le Tableau 4.2 sont fournis avec la distribution Linux-ATM de l'EPFL actuelle.

---

<sup>8</sup>La GPL est une sorte d'anti-licence qui assure la gratuité d'un logiciel et de ceux qui en découlent. Elle peut être caractérisée par la phrase : *"This program is free software ; you can redistribute it and/or modify it under term of the GNU General Public License as published by Free Software Foundation"*. La GPL se trouve à : <http://www.fsf.org/copyleft/copyleft.html>

Service	Description
<b>AAL0</b>	Accès direct à la commutation ATM (absence d'AAL)
<b>AAL5</b>	Accès direct au service de transport ATM
<b>Classical IP</b>	Avec encapsulation NULL ou LLC/SNAP et client/serveur ATMARP
<b>LAN Emulation</b>	Avec client LEC
<b>AREQUIPA</b>	<i>Application REQuested IP over ATM</i> , développée par l'EPFL pour établir des VC avec QoS entre deux hôtes IP directement connectés à ATM

Tab. 4.2 Les services supportés par le *driver* de l'EPFL

## Multicast sur Linux-ATM

La signalisation de la distribution Linux-ATM ne gère pas les SVC point-à-multipoint. Seule l'utilisation des PVC permet de réaliser le multicast au niveau ATM. Cette fonctionnalité est réalisée par configuration des VPI/VCI au niveau du commutateur ATM. Il suffit que l'hôte connaisse le VPI/VCI associé à la connexion pour envoyer et recevoir des données sur un PVC.

La mise en œuvre de PVC, et donc de configurations statiques, limite l'utilisation possible du service point-à-multipoint à des situations très stables ou expérimentales.

Le document [Oos97] indique l'importance du travail à réaliser pour utiliser le multicast sur CLIP avec la distribution Linux ATM de l'EPFL. En particulier, la gestion de la signalisation UNI 3.x doit être complétée avec les fonctionnalités point-à-multipoint. Le protocole MARS doit être intégralement implémenté côté client et serveur (il n'existe pas de version non commerciales de MARS actuellement). Le service de création des VC point-à-multipoint de type maillage doit aussi être réalisé dans les hôtes, avec accès aux paramètres de QoS pour l'utilisation de RSVP (la mise en oeuvre du type MCS est nettement plus complexe, en particulier avec la QoS).

### 4.5.2 La plate-forme Mirihade

Nous disposons pour nos expérimentations au sein du LIP6 d'un réseau ATM local. Celui-ci était relié à la plate-forme nationale MIRIHADÉ [ORC<sup>+</sup>98] au moment de nos expérimentations.

#### Présentation générale de Mirihade

La plate-forme nationale MIRIHADÉ est basée sur un réseau ATM longue distance reliant cinq plates-formes régionales. Celles-ci sont composées des centres de services et de laboratoires. Nous avons travaillé en particulier sur des connexions longue distance reliant le LAAS (Toulouse) et l'INRIA (Sophia-Antipolis) au LIP6 (Paris).

Les liens utilisés sont des PVP semi-permanents de la classe CBR avec un PCR à 10 Mbps.

Au niveau de la plate-forme parisienne, le LIP6 est associé à deux centres de services : l'UREC (Unité Réseaux du CNRS) et le CCR (Centre de Calcul). L'accès au réseau public se fait par des commutateurs ATM GDC fournis par France Télécom. La plate-forme ATM locale installée au LIP6 est composée d'un commutateur ATM Centillon 100 de Baynetwork et de plusieurs stations PC sous GNU/Linux et Sun sous Solaris (avec capacités multimédia). L'accès du LIP6 à MIRIHADÉ se fait à travers un commutateur Cisco LS1010 de l'UREC.

La plate-forme MIRIHADÉ nous fait donc bénéficier d'un environnement hétérogène au niveau du matériel réseaux, des stations et des systèmes d'exploitations utilisés.

### **Le multipoint sur Mirihade**

Le service multipoint des réseaux ATM utilise le mode connecté. Un échange de signalisation doit établir les connexions avant leur utilisation. Un premier problème apparaît ici : le temps de création de VC point-à-multipoint risque d'être très long, de part l'utilisation d'une signalisation initiale et surtout des mécanismes sous-jacents à cette signalisation.

Nous avons essayé d'évaluer ce délai initial au sein de la plate-forme MIRIHADÉ. Les fonctionnalités point-à-multipoint dynamiques (SVC) n'étant pas disponibles, nous l'avons estimé à partir de mesures sur des connexions point-à-point et d'observations générales nous permettant d'interpoler le temps de latence pour l'initiation des connexions point-à-multipoint à partir de celui des connexions point-à-point.

### **Les mesures sur Mirihade**

Sur la plate-forme MIRIHADÉ, nous avons effectué des mesures entre Paris et Toulouse. La Figure 4.19 indique la configuration utilisée. Les machines représentées sont directement connectées au réseau ATM. Nous avons trouvé un temps d'initiation moyen de l'ordre de 400 ms pour des connexions point-à-point entre stations PC sous GNU/Linux avec le *driver* de l'EPFL, le délai moyen de transit passant ensuite à 10 ms.

Des mesures moyennes dans les réseaux ATM longue distance nord-américains ont été rapportées dans [NBM<sup>+</sup>97]. Celles-ci indiquent des temps de latence pour la création de connexions point-à-point de l'ordre de 75 à 100 ms par lien entre noeuds intermédiaires actifs, ce qui semble correspondre aux valeurs trouvées sur MIRIHADÉ (cinq liens actifs entre deux stations terminales à Paris et Toulouse).



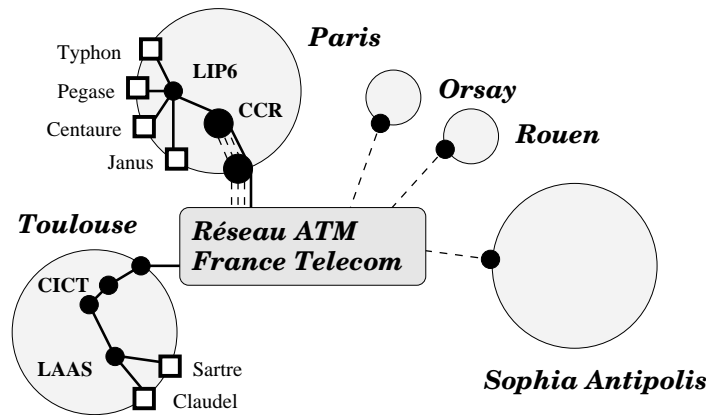


Fig. 4.19 Mesures sur la plate-forme MIRIHADÉ

<i>Liens actifs</i>	<i>Nombre de récepteurs</i>	<i>Délais estimés</i>	<i>Description</i>
5	1	1 secondes	1ère liaison (MIRIHADÉ)
5	10	10 secondes	10 récepteurs (MIRIHADÉ)
10	100	3 minutes	Vidéo-conférence
20	10000	10 heures	Diffusion vidéo

Tab. 4.3 Temps de création des VC point-à-multipoint

### Évaluation du délai de création de VC multipoint

Le rapport [NBM<sup>+</sup>97] ayant des résultats en accord avec nos mesures pour les connexions point-à-point et celui-ci s'appliquant à des réseaux ATM équivalents aux nôtres (réseaux ATM hétérogènes avec les mêmes types de commutateurs), nous avons interpolé les résultats pour notre plate-forme en ce qui concerne le multipoint. La valeur annoncée pour le point-à-multipoint est de 200 à 300 ms par élément intermédiaire **et** par récepteur impliqué dans la connexion. Ce résultat est lié à la sérialisation du traitement des créations de connexions point-à-multipoint dans les commutateurs actuels. Dans notre cas, avec par exemple une machine de Paris vers deux de Toulouse, nous nous situons dans un intervalle de 2 à 3 secondes pour le délai de création. Le temps d'initialisation de VC point-à-multipoint est donc très important pour quelques récepteurs. La mise en place de configurations envisagées pour la diffusion vidéo vers des milliers de récepteurs nous semblent difficilement réalisable dans ce contexte.

Le Tableau 4.3 présente les estimations de temps de création de VC point-à-multipoint. Nous y indiquons le nombre de liens actifs moyen entre l'émetteur et chaque récepteur ainsi que le nombre de récepteurs impliqués dans la connexion. Les deux premières lignes se situent dans l'environnement de la plate-forme MIRIHADÉ. Elles présentent le temps de création d'une liaison et de la connexion complète vers 10

émetteurs. Les deux lignes suivantes sont données à titre indicatif pour deux utilisations envisagées des réseaux ATM. La première présente l'exemple d'une vidéoconférence sur un réseau ATM de taille importante, et la dernière, l'exemple d'une diffusion vidéo à travers un réseau ATM très étendu (tel un réseau public avec service de SVC). Le temps de création des connexions devient inacceptable, or il apparaît à chaque renégociation dans les réseaux ATM actuels.

Les temps de latence que nous venons de présenter sont à relativiser selon l'utilisation des réseaux ATM. Dans notre cas, avec le modèle MARS, la portée d'une connexion ATM est limitée au *cluster* auquel appartiennent les participants.

### 4.5.3 Application Vidéo

Nous proposons d'expérimenter notre modèle à l'aide d'une application de diffusion vidéo utilisant un codage hiérarchique. Nous avons entrepris cette réalisation pour des PC sous GNU/Linux [Cal97] directement connectés à notre plate-forme ATM et disposant de caméras de type QuickCam. Le but de cette réalisation est de montrer un service multipoint, dynamique et hétérogène en terme de QoS, et s'appuyant sur un réseau ATM longue distance.

L'application, appelée **LaViD** (*L*ayered *V*iDéo), est composée de deux éléments : l'émetteur et le récepteur. L'application émettrice récupère le signal vidéo produit par une caméra et le transmet, selon un codage hiérarchique, sur des groupes IP multicast convenus à l'avance. La combinaison du trafic de ces groupes multicast est associée à des niveaux de QoS, tel que nous l'avons présenté dans la section 4.3.2.

La Figure 4.20 présente schématiquement notre expérimentation. L'image vidéo du personnage filmé par une caméra est, dans cet exemple, divisée en trois flux hiérarchiques par l'application émettrice située sur la machine H0. Ces flux sont distribués aux récepteurs en fonction de leurs demandes. Dans l'exemple, H1 a demandé les deux premiers flots et reconstitue donc une image de meilleure qualité que H2 qui ne reçoit que le premier. H3 peut être soit un récepteur, soit un routeur alimentant d'autres récepteurs situés hors du nuage ATM.

La Figure 4.21 présente l'interface graphique réalisée pour LaViD. Dans la partie gauche de la figure se trouve l'interface de l'émetteur où l'on peut régler le nombre de niveaux à émettre ainsi que divers paramètres de réglage de l'image. Dans la partie droite est présenté un récepteur avec sa fenêtre vidéo et l'interface spécifiant le nombre de couches à recevoir pour obtenir la QoS désirée. La notion de QoS est dans notre cas limitée à la qualité de l'image émise. La fréquence des images, et donc les paramètres du flux vidéo généré, sont liés à la capacité de la machine utilisée (celle-ci est limitée principalement par la capacité de la liaison entre la caméra et la station, et ici par le débit de la liaison parallèle). L'utilisateur influe donc indirectement sur la QoS.

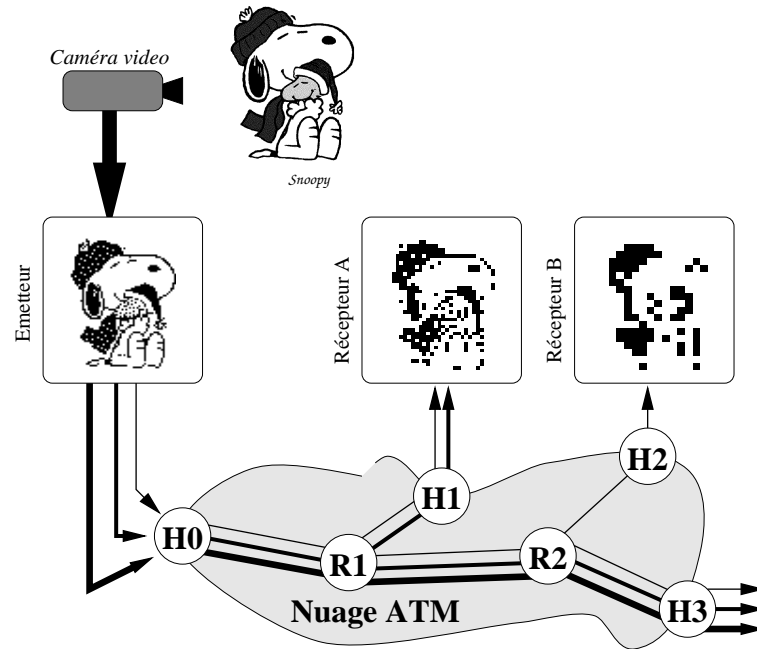


Fig. 4.20 Principe de l'expérimentation vidéo

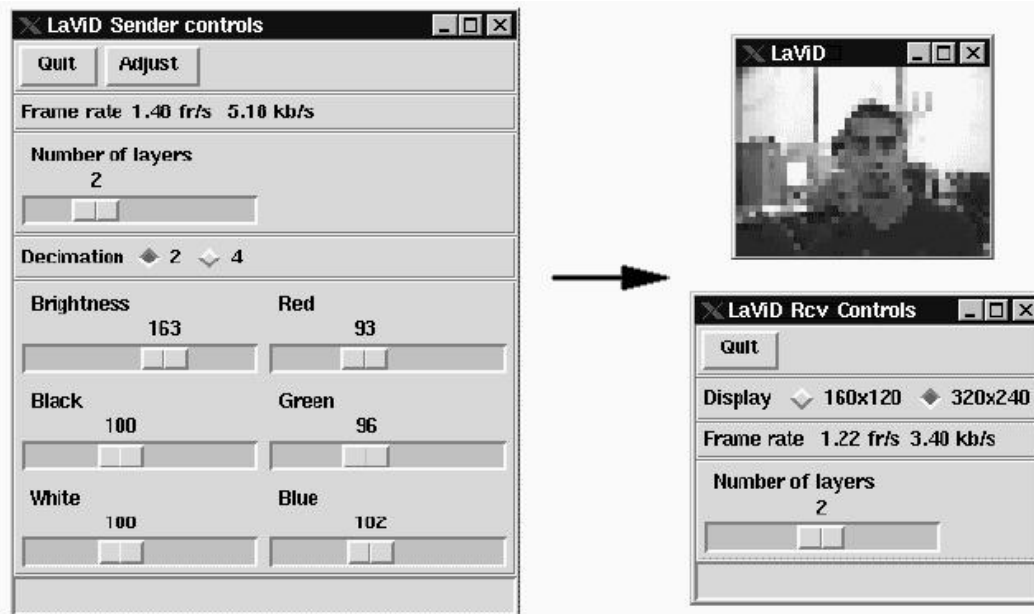


Fig. 4.21 Interface de l'application LaViD

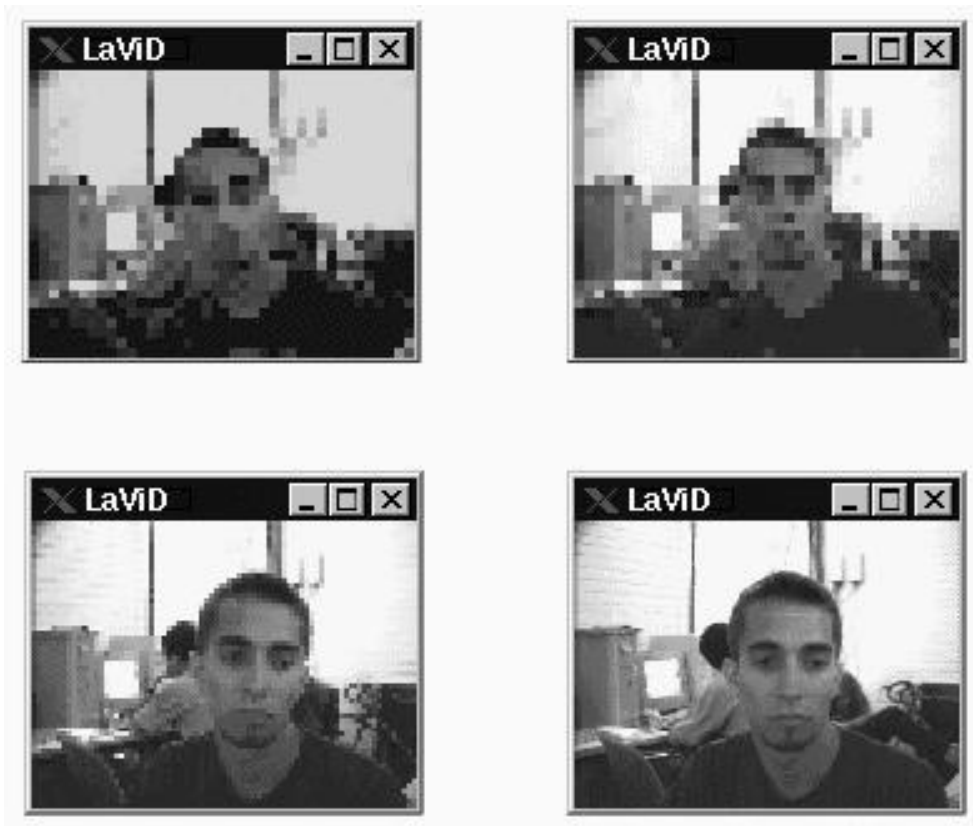


Fig. 4.22 Différentes qualités obtenues avec LaViD

La Figure 4.22 présente différents niveaux de QoS sélectionnés par le récepteur, à partir de l'émission de quatre couches. De gauche à droite et de haut en bas, nous pouvons observer quatre qualités différentes :  $40 \times 30$  pixels sur 4 plans (16 couleurs ou niveaux de gris) ;  $40 \times 30$  pixels sur 12 plans (4096 couleurs) ;  $80 \times 60$  pixels sur 12 plans ; et  $160 \times 120$  pixels sur 12 plans. L'application pouvant au maximum, avec cinq couches, afficher une image avec de une résolution de  $320 \times 240$  pixels sur 12 plans.

LaViD se place dans un environnement de prototypage. Nous n'utilisons pas RSVP et nous nous assurons seulement qu'il y ait une bande passante suffisante. La notion de QoS est réalisée par l'utilisation statique de connexions ATM dédiées pour simplifier notre réalisation. L'application est d'abord validée avec UDP/IP sur réseau Ethernet. Le passage sur ATM s'effectue en utilisant directement le service fourni par l'AAL5. Ne pouvant pas expérimenter sur des SVC point-à-multipoint (voir la section 4.5.1), nous avons basé notre validation sur des PVC point-à-multipoint, évitant ainsi tous les problèmes liés aux logiciels d'adaptation du multicast IP sur ATM non encore disponibles. Nous n'utilisons pas non plus CLIP, le but étant de montrer que l'on peut réaliser un service multicast dynamique et hétérogène sur les connexions ATM point-à-multipoint.

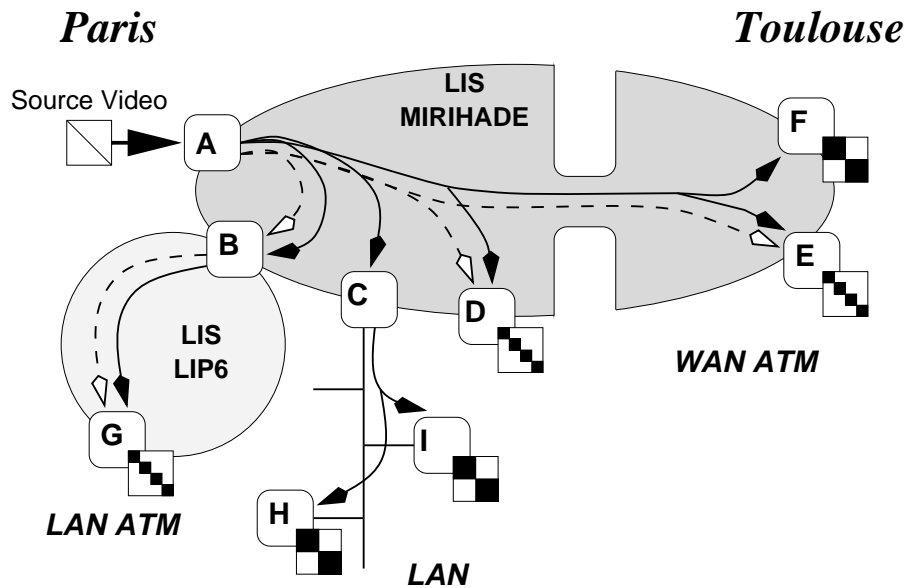


Fig. 4.23 Expérimentation sur MIRIHADÉ

### Adaptation à Mirihade

La Figure 4.23 illustre les possibilités de test de l'application vidéo envisagée sur MIRIHADÉ. L'exemple est donné avec deux niveaux de QoS obtenus par un codage hiérarchique avec deux couches. Les récepteurs D, E et G reçoivent les deux couches alors que F, H et I n'en reçoivent qu'une. Les trois environnements de test accessibles sur notre plate-forme sont représentés à travers des liens ATM locaux (LAN ATM avec le récepteur G sur notre plate-forme locale), des liens ATM de campus (le récepteur D directement connecté à notre centre de services), des liens ATM longue distance (WAN ATM avec les récepteurs E et F situés à Toulouse) et des liaisons mixtes (ATM et LAN en utilisant nos réseaux locaux classiques). L'application LaViD utilise, dans son stade de développement actuel, un accès direct à ATM (les configurations mixtes n'ont pas été réalisées).

### Configuration des PVC

Nous proposons d'expérimenter notre application LaViD sur une configuration de PVP établie manuellement. L'application émettrice récupère le signal vidéo et l'émet selon un codage hiérarchique fixé à cinq niveaux, sur cinq groupes IP multicast. L'émetteur et les récepteurs sont reliés directement à ATM.

Le maillage multicast réalisé entre Paris et Toulouse est présenté dans la figure 4.24. Nous avons une source à Paris qui émet vers cinq récepteurs dans cinq PVC correspondant aux cinq niveaux du codage hiérarchique. Les cinq récepteurs sont en fait quatre retours vers la machine source (ou ses voisines pour ne pas trop la charger)

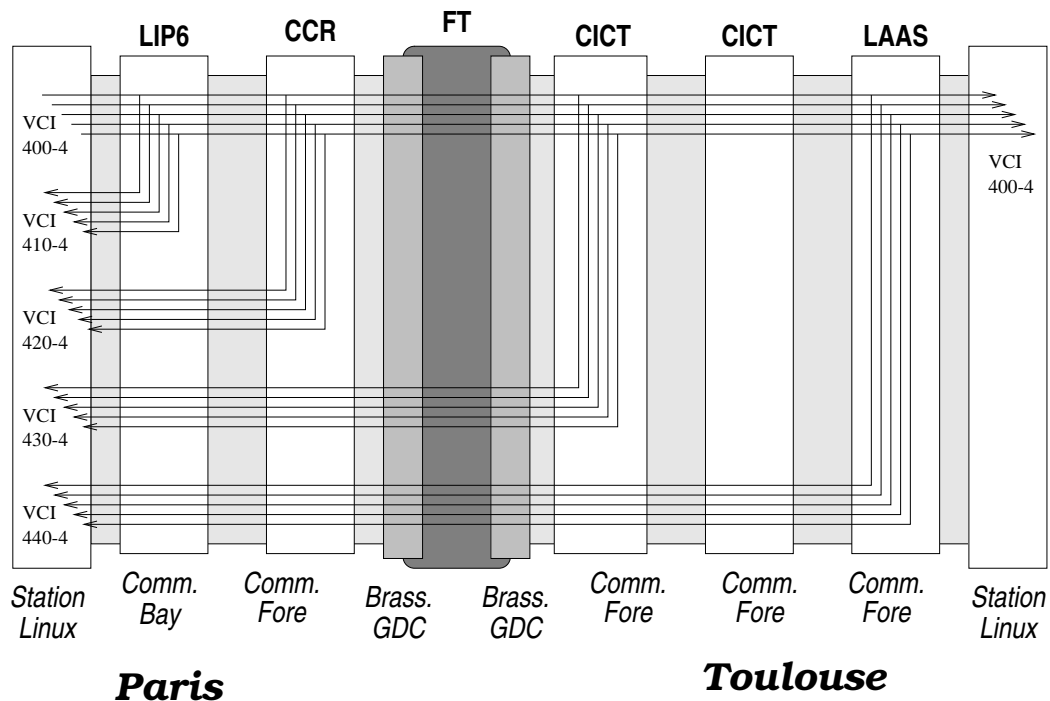


Fig. 4.24 Schéma de câblage des PVC point-à-multipoint

permettant ainsi de visualiser directement le comportement des récepteurs distants par rapport au signal de la source. Le cinquième récepteur se situe au LAAS pour une visualisation distante. Les numéros de VCI indiqués sont ceux fixés pour l'expérimentation, qui a ainsi nécessité la configuration de quatre commutateurs intermédiaires : celui de notre laboratoire (Bay – LIP6), ceux des centres de services (Fore – CCR, Fore – CICT) et celui du LAAS (Fore – LAAS). Les commutateurs installés par France Télécom (Brasseurs GDC – FT), tout le parcours chez l'opérateur et un commutateur du CICT manipulent des VP et sont transparents à nos expérimentations.

#### 4.5.4 Les résultats

L'utilisation de l'application LaViD en configuration statique nous a permis d'observer un service multipoint hétérogène et dynamique avec un temps de latence très faible. Les délais sont similaires à ceux rencontrés avec l'application LaViD utilisant UDP sur réseau Ethernet. Ce résultat est dû au fait que le temps d'une renégociation du niveau de QoS est principalement celui lié au traitement dans l'hôte pour l'accès au nouveau flux. Deux types de renégociation peuvent être réalisés :

- Au niveau de l'**émetteur**, permettant ainsi de constater les décalages des changements de QoS sur les divers récepteurs (la différence de délais maximum

étant de l'ordre de 20ms, le décalage des récepteurs n'est pas perceptible pour l'utilisateur).

- Au niveau des **récepteurs**, permettant d'observer un service hétérogène conforme à celui fourni par RSVP.

Globalement, les flux vidéos visualisés semblent l'être en local, alors qu'un réseau ATM effectuant un aller retour entre Paris et Toulouse est emprunté et que des temps de latence de l'ordre de plusieurs secondes seraient attendus en utilisant une renégociation avec la signalisation classique et que l'hétérogénéité serait absente. Ce type de résultat n'aurait pas été possible en renégociant une connexion ATM point-à-multipoint vers cinq récepteurs à chaque fois.

Nous n'avons pas effectué de campagne de mesures approfondies pour deux raisons. La première est que la configuration des connexions point-à-multipoint étant statique, les délais observés sont principalement liés au système d'exploitation et non au réseau. La seconde raison est que l'application LaViD a plutôt un but démonstratif et que sa réalisation confirme qu'un service multipoint avec QoS est réalisable.

Un autre aspect intéressant de la mise en œuvre de LaViD, est la première utilisation du multipoint sur réseaux ATM dans le cadre des tests de la plate-forme MIRIHADÉ. La mise en œuvre de cette fonctionnalité est assez lourde au moment de l'expérimentation. La disponibilité prochaine des SVC point-à-multipoint dans notre *driver* ATM, ainsi que de solutions MARS, devrait permettre la mise en place plus aisée de notre proposition.

Cette contribution reste d'actualité au niveau de l'approche et des choix technologiques réalisés. Elle peut s'appliquer pour résoudre de manière générale les problèmes liés aux réseaux maillés sur lesquels l'adaptation d'IP pose des problèmes d'efficacité. Par exemple, elle reste valable pour les architectures basées sur MPOA, si RSVP est envisagé lorsque MPOA sera plus mature [KKK<sup>+</sup>98].

## 4.6 CONCLUSION DU CHAPITRE

L'utilisation de multiples arbres multipoints pour transporter différents paliers de QoS permet d'assurer l'hétérogénéité des récepteurs tout en tirant parti des possibilités de connexions point-à-multipoint ATM. Notre proposition vise ainsi à fournir un service multipoint avec QoS sur un réseau commuté utilisant une signalisation orientée connexion.

Les limitations qui sont apparues avec les expérimentations, ne sont pas liées à notre approche mais plutôt à la gestion du multipoint associée à la signalisation ATM. En effet, le temps de latence pour la création d'un VC, déjà important pour du point-à-point, croît linéairement en fonction du nombre de récepteurs dans les commutateurs actuels. Cette caractéristique rend le service multipoint ATM impropre à

une utilisation comme support du service IP multicast, celui-ci devant soutenir des groupes de tailles importantes dynamiquement.

Pour une utilisation dès à présent, notre modèle avec pré-réservations pallie le temps d'établissement des VC ATM pour des groupes de petite taille, et fournit ainsi une dynamique conforme au modèle RSVP. Le besoin d'un service multipoint efficace et adapté aux grands groupes va motiver l'adaptation de la signalisation ATM, mais ces modifications vont certainement demander beaucoup de temps. La dernière version de l'interface ATM, UNI 4.0 devrait, avec sa fonctionnalité LIJ, améliorer les performances du modèle sans pré-réservation. Mais cela n'évitera pas la sérialisation des requêtes qui est un problème d'implémentation de la signalisation au sein des commutateurs. Ces temps de création de connexions point-à-multipoint importants empêchent actuellement l'utilisation d'un service multipoint s'adaptant au facteur d'échelle.

Les limitations du service multipoint d'ATM pourraient être réduites en utilisant une signalisation avec création parallèle des branches d'une connexion multipoint, à condition de disposer de la puissance nécessaire au niveau de l'émetteur. Mais nous serions toujours en mode connecté avec les limitations liées au temps de latence initial important. Ce problème ne disparaît qu'en utilisant un mode non connecté. C'est ce qui nous a motivé pour trouver une autre approche pour l'adaptation d'IP sur ATM, sans la limitation majeure précitée. Dans le prochain chapitre, nous proposons de ne pas exploiter de signalisation orientée connexion, en tirant parti d'une approche similaire à la Commutation IP que nous considérons comme une alternative prometteuse.





## CHAPITRE 5

# *Les services multipoints avec la Commutation par Label*

### Contenu du chapitre

*Ce chapitre présente notre seconde contribution. Nous nous intéressons à l'intégration du multipoint avec QoS sur les réseaux commutés avec signalisation en mode non connecté. Nous présentons tout d'abord notre solution "RSVP Switching" et proposons une approche générique pour gérer les fonctionnalités de la couche réseau au niveau de la commutation. Nous développons particulièrement l'adaptation des mécanismes d'agrégation et de filtrage de RSVP. Nous terminons en proposant une implémentation de notre solution basée sur IPv6 dans un environnement ATM.*

*Ce chapitre est organisé de la manière suivante :*

- ⇒ Section 5.1 – Introduction du chapitre – p. 108.
- ⇒ Section 5.2 – La proposition RSVP Switching – p. 109 : *Choix d'une approche de commutation. Intégration de QoS dans IP Switching. Présentation de notre proposition.*
- ⇒ Section 5.3 – L'intégration du service multipoint de RSVP – p. 112 : *Description de la réalisation d'un "cut-through" avec RSVP Switching. Description des fonctions de filtrage et d'agrégation.*
- ⇒ Section 5.4 – Spécialisation du modèle RSVP Switching – p. 120 : *Spécialisation de notre modèle à IPv6 et à la commutation ATM.*

- ⇒ Section 5.5 – Comparaison avec d’autres intégrations – p. 124 : *Présentation d’autres approches de la commutation IP avec QoS telles que IS-IPSO, MPLS ou QSR, et comparaison avec RSVP Switching.*
- ⇒ Section 5.6 – Modification de RSVP – p. 125 : *Ajout de la gestion des labels dans RSVP et modification de l’interface pour servir RSVP Switching.*
- ⇒ Section 5.7 – Réalisation d’un RSVP Switch – p. 129 : *Présentation de la plateforme pour l’implémentation de RSVP Switching. Émulation de la fonction de commutation. Intégration de service dans RSVP Switching*
- ⇒ Section 5.8 – L’expérimentation – p. 136 : *Présentation des applications de DIS et de leur extension pour RSVP. Expérimentation de RSVP Switching avec les applications de DIS.*
- ⇒ Section 5.9 – Conclusion du chapitre – p. 140.

## 5.1 INTRODUCTION DU CHAPITRE

Dans l’optique de fournir un service multipoint avec QoS dans un environnement commuté avec une signalisation orientée connexion, nous avons développé, dans le chapitre précédent, une solution basée sur l’extension du modèle *Classical IP* (CLIP). Cette solution met en évidence les difficultés liées à l’utilisation du mode connecté de la signalisation des réseaux ATM pour une gestion efficace du multipoint avec QoS. Nous avons donc recherché une approche différente pour intégrer les nouvelles fonctionnalités dans les réseaux commutés.

Nous avons présenté, dans le chapitre 3, un troisième type de solution pour l’intégration d’IP sur ATM qui évite le mode connecté des réseaux ATM. Cette approche, appelée “Commutation IP” dans le cadre de l’environnement IP sur ATM, repose sur la commutation associée à une signalisation en mode non connecté. Elle a pour but de simplifier l’intégration des deux technologies tout en profitant au mieux des performances de la commutation et en conservant la richesse fonctionnelle du niveau routage.

La Commutation IP recouvre de nombreuses solutions, qui ne sont pas forcément spécifiques à IP et ATM, mais qui sont basées sur les mêmes principes : **échange de label** (*label swapping*) dans les nœuds intermédiaires du réseau et **commutation de paquets** (*packet switching*) basée sur l’association rapide de labels permettant ainsi d’éviter le chemin de routage habituel (utilisation d’un chemin direct au niveau 2, appelé *cut-through*, dans les nœuds). Le tout forme l’approche générique appelée “**Commutation par Label**” (*Label Switching*).

Les solutions proposées dans le cadre de la Commutation par Label intègrent généralement le multipoint mais sans QoS. Pour pallier ce manque, nous adoptons par la suite une approche dans laquelle nous adaptons le protocole RSVP à la Commutation par Label. Cette approche nous permet, outre un gain en performance

significatif, une intégration efficace des nouvelles fonctionnalités grâce au couplage simplifié d'IP et ATM évitant le modèle avec serveur. La solution que nous proposons s'appelle "**RSVP Switching**" et met en œuvre les mécanismes nécessaires au fonctionnement du modèle multipoint-à-multipoint hétérogène et dynamique du service IP multicast avec QoS dans un environnement commuté. Nous spécialisons ensuite notre approche pour un usage avec IP sur ATM et proposons une intégration au sein de la plate-forme SAFIR pour réaliser des expérimentations à l'aide d'une application de DIS aux besoins en QoS importants.

L'approche théorique de cette contribution a été publiée dans la revue *Telecommunication Systems* [FF98]. Son extension expérimentale est associée au projet DIS/ATM et a donné lieu à un rapport [FRDF98].

## 5.2 LA PROPOSITION RSVP SWITCHING

Les solutions basées sur la Commutation par Label se répartissent en deux catégories principales : la première est dite "orientée flot" (*flow driven*) et comprend les solutions qui se basent sur le trafic de données local pour décider de l'utilisation de labels ; la seconde est dite "orientée topologie" (*topology driven*) et regroupe les solutions basées sur une distribution systématique des labels selon la signalisation de routage. Pour fournir des garanties de QoS, la notion d'état doit être introduite au sein du réseau en maintenant des informations dans les éléments intermédiaires. Le protocole RSVP est adapté pour maintenir ces informations dans chaque nœud pour l'environnement IP. En conséquence, nous intégrons ce protocole dans l'approche de la Commutation par Label, créant ainsi une troisième catégorie de solutions basées sur l'association des labels avec les flots et les sessions RSVP. Cette approche est dite "orientée session" (*session driven*) et permet la gestion de la QoS.

### 5.2.1 Le choix d'une approche

RSVP peut être introduit dans les deux premières catégories (orientées flot ou topologie). Il produit les solutions "orientées session" suivantes :

- L'introduction de RSVP dans l'approche "orientée flot" est naturelle, car cette approche est conçue dans l'optique d'identifier les flots localement après observation du trafic. Avec RSVP, l'approche utilise aussi les flots mais ceux-ci sont identifiés à partir de la signalisation RSVP.
- L'approche "orientée topologie", initialement présentée pour le trafic unicast et basée sur les informations de routage, a été étendue pour tenir compte de tout type de signalisation, ce qui en fait une approche "orientée signalisation" (*control driven*). Dans ce cadre, RSVP est l'un des protocoles de signalisation pour le transport des labels, au même titre que les signalisations de routage et du multicast.

Même si ces deux approches “orientées session” se comportent sensiblement de la même manière, toutes deux utilisant la signalisation RSVP pour transporter des informations sur les labels associés au flots, on peut noter que la gestion de RSVP s’intègre naturellement dans l’approche “orientée flot”. Avec l’approche “orientée signalisation”, chaque protocole est intégré séparément ce qui produit un modèle plus disparate.

Des différences importantes existent aussi au niveau de la gestion du multicast (voir la section 3.4.3), les approches “orientées flot” l’intégrant simplement au même titre que l’unicast.

Ainsi, c’est l’approche “**orientée flot**” dans laquelle nous avons décidé d’intégrer le multicast IP avec QoS, en y adjoignant une architecture RSVP. Nous l’avons choisie pour sa plus grande cohérence par rapport aux autres solutions. Nous nous sommes basés pour cela sur la solution **IP Switching** pour laquelle ce genre d’extension n’existe pas et avons appelé notre proposition “**RSVP Switching**”.

### 5.2.2 IP Switching et QoS

La solution IP Switching (voir la section 3.4.1), basée sur l’approche “orientée flot”, repose sur le principe de localité pour respecter le principe d’absence d’état de l’Internet. Dans les protocoles de gestion du commutateur [NEH<sup>+</sup>98] proposés, l’introduction de niveaux de services différents est prévue, allant des débits fixes au WRR (*Weighted Round Robin*) en passant par les niveaux de priorité, mais uniquement pour une utilisation statique. Les performances annoncées (gestion de plus de 100.000 files d’attente au niveau du commutateur) indiquent que le matériel est à la hauteur d’une intégration de la QoS.

L’introduction de RSVP dans IP Switching permet de combler l’absence d’architecture pour réaliser une gestion de la QoS dynamique. C’est une étude de ce type d’intégration que nous proposons de faire dans la suite du chapitre.

### 5.2.3 Notre proposition

Notre but est de fournir le service IP multicast avec QoS en utilisant RSVP au sein d’une architecture basée sur la Commutation par Label. Nous proposons donc une solution générique que nous spécialiserons à IP et à ATM dans la suite. La base de notre réflexion est la solution “orientée flot” **IP Switching** où les décisions locales de commutation sont guidées par la signalisation RSVP plutôt que par le trafic observé. Nous utilisons donc une approche “orientée session” où RSVP est utilisé pour la gestion des réservations qui sont associées à la création des chemins commutés.

Nous appelons notre solution **RSVP Switching** car nous ne commutons que les flots associés à la QoS. Le trafic *Best Effort* est simplement routé, mais des mécanismes

	Niveau routage	Niveau commutation
<i>État</i>	Dynamique (Soft State)	Dynamique (Soft State)
<i>QoS Multipoint</i>	Hétérogène (filtrage)	Homogène (matrice de commutation)
<i>Modèle Multipoint</i>	N-vers-M (agrégation)	1-vers-M (matrice de commutation)

**Tab. 5.1** Comparaison des niveaux routage et commutation

identiques à ceux de IP Switching peuvent aussi être mis en œuvre. Une limitation des solutions basées sur l’approche “orientée flot” est due au fait que l’efficacité dépend de la proportion de données commutées. Avec une approche basée sur RSVP, ce problème disparaît car tout le trafic RSVP est commuté.

Nous pensons que le choix de l’approche “orientée flot” est plus efficace pour son intégration naturelle des trafics unicast et multicast. De même, conserver les données avec la signalisation au sein du parcours routé usuel avant de procéder à la commutation permet une utilisation transparente de RSVP. L’utilisation du multicast au niveau commutation est prévue avec les flots RSVP. Les problèmes qui lui sont liés seront résolus ensembles. Le trafic IP multicast *Best Effort* est supposé routé. Si une commutation de type IP Switching est envisagée pour ce trafic, les problèmes non résolus tels que l’agrégation sont à prendre en compte.

Au niveau de la commutation, la matrice nous fournit localement des connexions point-à-multipoint avec une QoS homogène alors qu’au niveau du routage, le modèle multipoint de RSVP supporte des connexions multipoint-à-multipoint hétérogènes. La Table 5.1 présente une comparaison des services fournis par les deux modèles au niveau d’un nœud. Nous supposons que la technologie de commutation a les mêmes limitations que celle d’ATM.

La ligne “*État*” de la Table 5.1 décrit la manière dont les informations sont conservées dans un élément du réseau. Les deux modèles s’appuient sur un état dynamique. Au niveau routage, le *Soft State* de RSVP est utilisé pour maintenir les réservations dans le réseau. Au niveau commutation, avec IP Switching, un mécanisme équivalent est mis en œuvre pour la distribution des labels à travers IFMP. Dans notre solution, nous utilisons directement RSVP à la place d’un protocole de distribution de label spécifique. Nous réutilisons ainsi le *Soft State* de RSVP et lui assignons la double fonction de gestion des réservations et des informations de commutation.

La ligne “*QoS Multipoint*” de la Table 5.1 précise si chaque port de sortie peut avoir une QoS différente. Au niveau routage, chaque destination peut choisir une QoS indépendamment. Cela provient du modèle orienté récepteur de RSVP où chaque récepteur choisit sa réservation indépendamment. Au niveau commutation, une simple duplication est effectuée, ne permettant qu’une QoS homogène sur les branches de

l'arbre de distribution. La fonctionnalité de filtrage utilisée au niveau routage pour fournir l'hétérogénéité n'est pas disponible au niveau commutation, créant ainsi un premier problème d'adaptation de RSVP.

La ligne "*Modèle Multipoint*" de la Table 5.1 présente le type de connexions multipoints disponible. Au niveau routage, IP multicast permet un modèle *N-vers-M*. RSVP utilise ce modèle et propose d'y ajouter différents types d'agrégation pour la gestion de la QoS. Cette fonctionnalité est, de même que le filtrage, non disponible au niveau de la commutation. Si rien n'est prévu au niveau commutation, l'agrégation de flots résulte en un multiplexage des données, qui dans le cas d'une technologie basée sur des cellules, entraîne la perte des informations.

La mise en œuvre des différents mécanismes nécessaires à la réalisation des fonctionnalités RSVP au niveau commutation dans **RSVP Switching** est présentée dans la suite.

### 5.3 L'INTÉGRATION DU SERVICE MULTIPOINT DE RSVP

Dans cette section, nous présentons les mécanismes de commutation mis en œuvre pour les flots RSVP, puis nous étudions particulièrement l'intégration de deux fonctionnalités qui posent des difficultés : le filtrage et l'agrégation.

#### 5.3.1 Cut-through

Le principe de base de la Commutation par Label est la création de *cut-through*, c'est-à-dire la commutation au lieu du routage usuel au niveau d'un nœud. Nous décrivons dans cette section la mise en place d'un *cut-through* au niveau d'un élément mettant en œuvre la solution RSVP Switching appelé un **RSVP Switch**. La Figure 5.1 présente un chemin de routage (*Routed path*) entre trois RSVP Switches pour la session **S** (le trafic remonte dans la couche 3 pour les trois éléments). Avant et après la zone définie par ces trois éléments, il peut y avoir d'autres RSVP Switches, des réseaux IP avec RSVP ou des hôtes directement connectés. Sur la figure, nous avons décomposé dans chaque RSVP Switch la couche réseau où se situent les fonctionnalités de routage avec RSVP (*Routing*) et la couche liaison où se trouve le moteur de commutation (*Switching*). Le trafic de signalisation est représenté par des flèches bidirectionnelles blanches (*Signaling traffic*) et celui des données par des flèches unidirectionnelles noires (*Data traffic*).

Après le passage du message PATH, les réservations se mettent en place. Les données sont séparées de la signalisation. Elles quittent le chemin routé par défaut pour suivre un chemin avec label représenté au niveau commutation. Les informations relatives aux labels sont transmises à travers les messages RESV, dans un nouvel objet spécifique ajouté à la signalisation RSVP dans ce but. Nous appelons cet objet "LABEL". Après avoir échangé des messages avec l'information Label (contenue dans

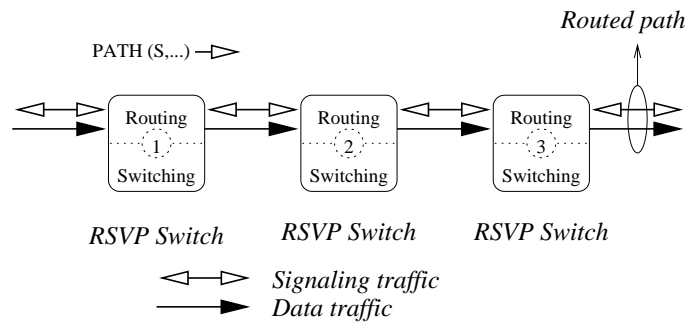


Fig. 5.1 Chemin de routage initial

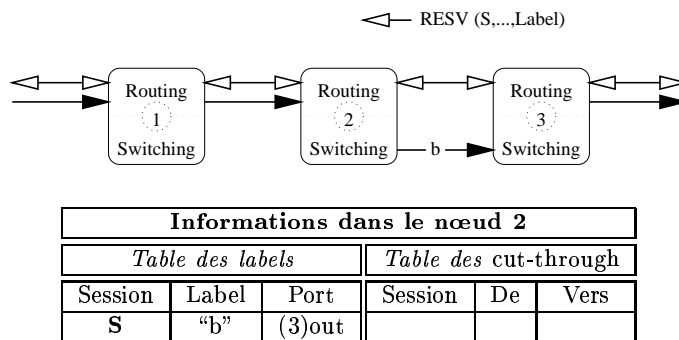


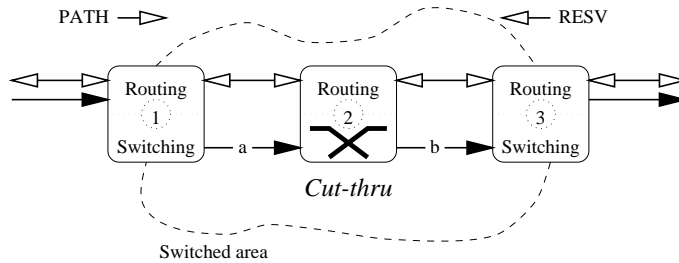
Fig. 5.2 Création du chemin avec label

l'objet LABEL), nous pouvons créer un lien virtuel spécifique étiqueté grâce au label. Des réservations distinctes correspondent à des liens différents.

Dans la Figure 5.2, un message RESV est propagé du récepteur vers l'émetteur. Le message RESV contient l'objet LABEL pour étiqueter le lien spécifique. Dans la partie gauche du dessin, entre les RSVP Switch 1 et 2, les données sont encore routées. Dans la partie droite, une réservation a été réalisée. La liaison correspondante est établie et les données, même si elles restent routées, appartiennent à un lien virtuel différent de celui du trafic routé. Les flots avec labels attendent pour l'établissement d'un *cut-through*. Le nouveau lien est représenté par une flèche au niveau de la couche liaison avec le label "b". Le label est local au lien considéré et change de lien en lien.

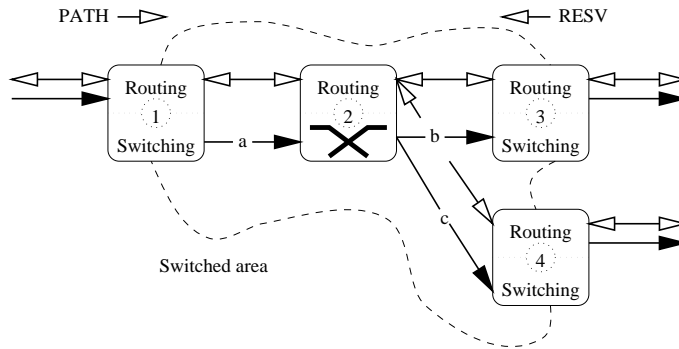
Dans la figure 5.3, un label "a" pour la session S a été attribué entre les RSVP Switches 1 et 2. Un *cut-through* peut être réalisé. Dans les tableaux associés aux figures, nous supposons qu'un lien ou un port du nœud N est référencé par la notation (N). Dans la table de commutation (ou "Table des cut-through") du RSVP Switch 2, les données arrivent du port d'entrée (1) avec le label "a" et sont commutées vers le port de sortie (3) avec le label "b" correspondant. A ce niveau nous ne considérons pas les problèmes liés au filtrage ou à l'agrégation.





Informations dans le nœud 2					
Table des labels			Table des cut-through		
Session	Label	Port	Session	De	Vers
S	"b"	(3)out	S	"a"(1)	"b"(3)
S	"a"	(1)in			

Fig. 5.3 Création d'un cut-through unicast



Informations dans le nœud 2					
Table des labels			Table des cut-through		
Session	Label	Port	Session	De	Vers
S	"b"	(3)out	S	"a"(1)	"b"(3)
S	"c"	(4)out		"a"(1)	"c"(4)
S	"a"	(1)in			

Fig. 5.4 Création d'un cut-through multicast

Dans la figure 5.4, un exemple de flot multicast est établi en adjoignant une nouvelle entrée dans la table de commutation. Cela est réalisé en ajoutant un message RESV qui produit l'association du label "c" pour ce flot en direction de (4).

Les frontières de la zone de commutation (*Switched area*) sont définies par les RSVP switches entrants ("*ingress*" *node*) et sortants ("*egress*" *node*). Ces éléments entrants ou sortants sont les points par lesquels les données commencent ou terminent leur passage dans la zone commutée. Dans la Figure 5.4, le nœud 1 est un élément entrant, les nœuds 3 et 4 sont des éléments sortants.

L'utilisation de la commutation a un impact sur l'en-tête des données. Par exemple le TTL n'est plus modifié tant que le paquet progresse dans la couche liaison sans remonter au niveau du routage. Nous devons donc le recalculer à partir des paquets de signalisation qui sont toujours routés. En conservant la signalisation au niveau routage, on évite les boucles de manière habituelle.

Les obstacles que nous présentons dans la suite sont identiques à certains que nous avons déjà rencontrés dans d'autres approches de l'intégration d'IP sur ATM. Ils sont liés au modèle restrictif de la couche liaison qui ne peut être étendu sans remonter dans une couche supérieure. Pour mettre en œuvre la solution RSVP Switching, nous devons déplacer les fonctionnalités qui devaient prendre place au niveau routage hors de la zone commutée. Ainsi le filtrage et l'agrégation, qui ne peuvent avoir lieu dans la couche liaison, sont réalisés en dehors de cette zone. C'est ce que nous présentons dans les deux prochaines sections.

### 5.3.2 Filtrage

Dans RSVP, la fonctionnalité de filtrage permet de sélectionner des sous-ensembles de paquets dans les flots d'une session. Ces sous-flots sont définis par n'importe quel champ des en-têtes d'IP et des couches supérieures. Par exemple, on utilise habituellement un filtrage par émetteur pour sélectionner les flots dans une session après une agrégation. D'autres types de filtrage permettent d'extraire différents sous-flots dans un flot utilisant un codage hiérarchique. Dans la version actuelle des spécifications de RSVP, seule l'adresse des émetteurs est utilisée pour des raisons de simplicité, mais l'extension peut être réalisée de manière aisée.

Dans la suite, nous présentons comment nous gérons le filtrage pour une session RSVP avec sous-flots. Nous proposons d'utiliser un "pré-filtrage" avant la zone commutée à l'aide de la signalisation de réservation. La signalisation continue d'emprunter le chemin routé et les messages PATH ne sont pas modifiés. L'arbre de retour est construit par ces messages de manière habituelle. Les messages RESV sont étendus pour transporter l'objet LABEL. Pour fournir la QoS hétérogène orientée récepteur, plusieurs arbres multicast sont créés au niveau de la couche liaison afin de recevoir le trafic de données commuté de chaque sous-flot. Ces arbres sont construits pendant

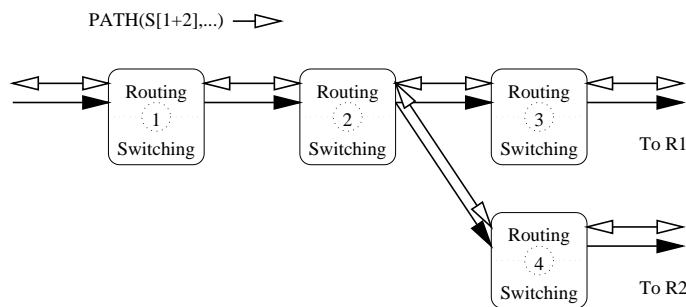


Fig. 5.5 Chemin initial routé pour le filtrage

la phase de réservation. Chaque sous-flot est caractérisé par une QoS homogène ainsi que le requiert la couche de commutation.

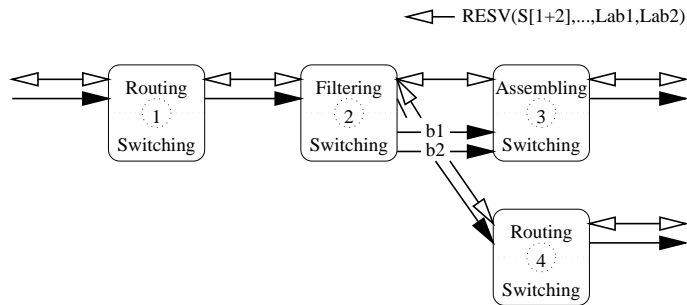
Dans le but d'éviter le problème du filtrage au niveau de la couche liaison, celui-ci est géré au début de la zone commutée. L'élément entrant doit assurer la fonctionnalité de filtrage en sélectionnant et transmettant les paquets de données sur l'arbre multicast correspondant. A la sortie de la zone commutée, un élément sortant peut avoir besoin de plusieurs sous-flots pour recréer le flot avec la QoS demandée (le filtrage maximum est toujours réalisé en entrée de la zone commutée pour préserver la dynamique du modèle).

Nous présentons dans la Figure 5.5, un émetteur générant un flot de données dans la session RSVP  $S$ . Ce flot est composé de deux sous-flots (1 et 2). Les récepteurs peuvent demander le flot complet ou un sous-ensemble à travers l'un des sous-flot. Les messages PATH transportent l'information vers les récepteurs pour leur permettre de choisir les sous-flots adéquats à demander lors de la phase de réservation.

Dans la Figure 5.6, le récepteur R1 demande le flot complet et effectue une réservation de ressources pour les sous-flots 1 et 2. Les RSVP Switches gèrent les liens virtuels en associant chaque sous-flot aux labels respectifs "b1" et "b2".

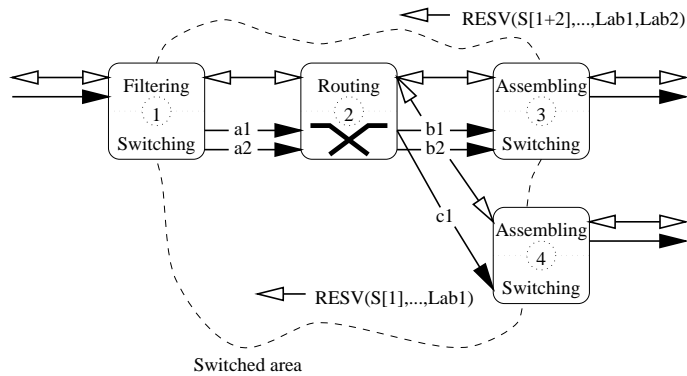
Dans la Figure 5.7, le récepteur R2 réclame une QoS plus faible en ne réservant des ressources que pour le sous-flot 1. Ce sous-flot est associé au label "c1". La réservation effectuée entre les nœuds 1 et 2 est le "maximum" des deux réservations précédentes (agrégation des messages RESV), ce qui correspond aux réservations pour les sous-flots 1 et 2. Les labels associés aux liens virtuels respectifs sont "a1" et "a2".

Les flots entrants et sortants du RSVP Switch 2 ont un label ; les *cut-through* peuvent être mis en place. Pour le sous-flot 1 de la session  $S$ , un *cut-through* multicast de (1) avec le label "a1" vers (3) et (4) avec les labels respectifs "b1" and "c1" est construit. Pour le sous-flot 2 de la session  $S$ , un *cut-through* unicast est créé de (1) avec le label "a2" vers (3) avec le label "b2".



Informations dans le nœud 2					
Table des labels			Table des cut-through		
Session	Label	Port	Session	De	Vers
S sub 1	"b1"	(3)out			
S sub 2	"c2"	(3)out			

Fig. 5.6 Association des labels pour le filtrage



Informations dans le nœud 2					
Table des labels			Table des cut-through		
Session	Label	Port	Session	De	Vers
S sub 1	"b1"	(3)out	S sub 1	"a1"(1)	"b1"(3)
S sub 2	"b2"	(3)out	S sub 2	"a1"(1)	"c1"(4)
S sub 1	"c1"	(4)out	S sub 2	"a2"(1)	"b2"(3)
S sub 1	"a1"	(1)in			
S sub 2	"a2"	(1)in			

Fig. 5.7 Création du cut-through pour le filtrage

La fonctionnalité de filtrage est assurée dans le nœud 1. Celui-ci est l'élément entrant de la zone commutée. Ce décalage du filtrage évite de le réaliser au sein du nœud 2 où il aurait eu lieu avec le routage. De même ce mécanisme permet d'utiliser le modèle multicast homogène dans la zone commutée en conservant globalement les possibilités de RSVP.

Dans le nœud sortant 3, les deux sous-flots sont nécessaires pour reconstruire le flot initial réclamé. Dans le nœud sortant 4, seul le sous-flot 1 est nécessaire. Les éléments sortants de la zone commutée sont responsables du calcul du nouvel en-tête des paquets, après que celui-ci ait été laissé inchangé après son passage continu dans la couche liaison.

### 5.3.3 Agrégation

L'agrégation est une fonctionnalité nécessaire au modèle RSVP pour permettre l'utilisation efficace du service multipoint-à-multipoint. Cela permet de réduire la consommation de ressources par rapport au service point-à-multipoint de la même manière que le multicast le fait par rapport à l'unicast. Associée avec le filtrage, l'agrégation permet de réunir des flots d'émetteurs différents au sein d'une même session pour les séparer ensuite.

Le problème posé par l'agrégation est difficile à contourner en environnement commuté à cause de l'absence de fonctionnalité multipoint-à-point au niveau de la couche liaison. La question à résoudre est comment, au niveau commutation, utiliser plusieurs labels vers un seul en sortie sans entrelacement. Nous utilisons des circuits séparés dans tout le parcours commuté. Nous utilisons le même principe que pour le filtrage, nous déplaçons la fonctionnalité hors de la zone commutée. L'agrégation a lieu après les *shortcuts* pour permettre à tout flot de rejoindre une session, et ce dans n'importe quel nœud. A la sortie de la zone commutée, dans l'élément sortant, les différents flots remontent au niveau routage où ils sont agrégés.

La séparation des flots ne signifie pas que nous séparons les réservations, et effectuons ainsi de la sur-réservation. Nous conservons les identifiants de la couche liaison (les labels) séparés, tout en groupant les ressources qui leur sont attribuées. Nous associons une réservation commune à plusieurs liens virtuels. La section 5.4.2 présente la manière de la réaliser avec la technologie ATM. Dans la couche réseau, nous conservons la signalisation RSVP qui nous indique une réservation pour la session, mais plusieurs objets LABEL sont transportés au sein des messages RESV pour identifier chaque sous-flot. Pour les styles RSVP, on obtient :

**WF et SE** (*Wildcard Filter* et *Shared Explicit*) Les styles de réservations partagées utilisent un label par émetteur et une seule réservation correspondant au "maximum" des ressources demandées. Le nombre d'émetteurs avec le style WF est potentiellement élevé, mais dans la pratique, il est limité par l'objet RSVP SCOPE.

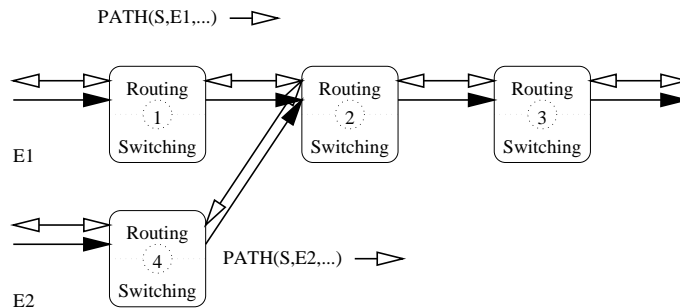
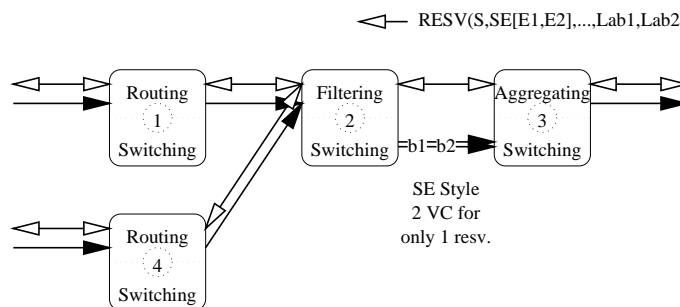


Fig. 5.8 Chemin initial routé pour l'agrégation

**FF** (*Fixed Filter*) Le style de réservations séparées utilise un label et une réservation par émetteur. La quantité de ressources demandées correspond à la “somme” des ressources réservées pour chaque flot.

Dans la Figure 5.8, deux émetteurs (E1 et E2) envoient des messages PATH dans la session **S**. Leurs trafics de données vers le récepteur R sont agrégés. Les messages PATH transportent l'information de leurs sources pour permettre le filtrage.

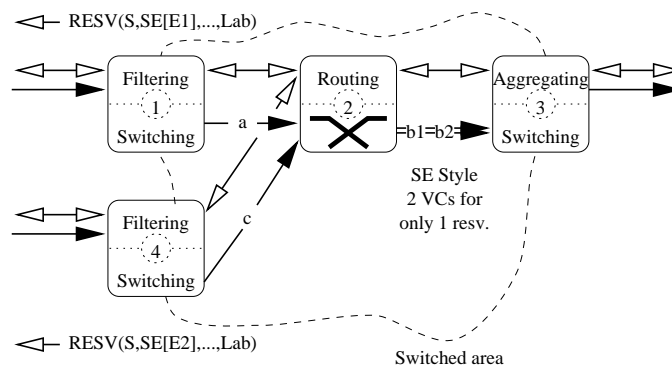


Informations dans le nœud 2					
Table des labels			Table des cut-through		
Session	Label	Port	Session	De	Vers
S(E1)	“b1”	(3)out			
S(E2)	“b2”	(3)out			

Fig. 5.9 Association des labels pour l'agrégation

La réponse aux messages PATH est un message RESV avec deux labels “b1” et “b2”, correspondant aux deux sources et aux flots associés dans la session **S** (Figure 5.9). Seule une réservation est gérée, mais elle est associée aux deux flots. Au niveau commutation, deux liens virtuels sont créés.

Lorsque l'affectation des labels est terminée de chaque côté du nœud 2 (Figure 5.10), le *cut-through* est créé. Le trafic de données de chaque émetteur utilise des labels différents dans toute la zone commutée, restant ainsi dans des circuits virtuels distincts



Informations dans le nœud 2					
Table des labels			Table des cut-through		
Session	Label	Port	Session	De	Vers
S(E1)	"b1"	(3)out	S (E1)	"a"(1)	"b1"(3)
S(E2)	"b2"	(3)out			
S(E1)	"a"	(1)in	S (E2)	"c"(4)	"b2"(3)
S(E2)	"c"	(4)in			

Fig. 5.10 Création du *cut-through* pour l'agrégation

et évitant le multiplexage au niveau de la couche liaison. Chaque flot est associé à un *cut-through* dans le nœud 2. Les flots sont agrégés à la sortie de la zone commutée.

La fonctionnalité d'agrégation est déplacée du nœud 2, où elle a lieu au niveau RSVP, vers le nœud sortant 3. Ce mécanisme autorise l'utilisation de RSVP dans un environnement commuté. Comme pour le filtrage, à la sortie de la zone commutée, il est nécessaire de recalculer l'en-tête des paquets.

L'agrégation et le filtrage sont réalisés dans chaque nœud intermédiaire de la zone commutée pour fournir le service dynamique associé à RSVP. La zone commutée peut être variable, donc tous les RSVP Switches doivent pouvoir assurer les fonctionnalités des nœuds entrants et sortants. Un nœud entrant gère la séparation en sous-flots et un nœud sortant assure le réassemblage partiel de ces sous-flots ainsi que l'agrégation. La gestion des sous-flots est identique à la gestion du style SE dans notre modèle. Les RSVP Switches sont aussi tenus d'assurer les fonctionnalités de base associées au protocole de routage, à l'architecture RSVP et aux mécanismes de commutation.

#### 5.4 SPÉCIALISATION DU MODÈLE RSVP SWITCHING

La solution RSVP Switching utilise une approche générique, en ne spécifiant pas de protocole de réseau particulier, ni de technologie de commutation, même si les limitations envisagées de cette dernière sont fortement typées commutation de cel-

lules. Nous présentons en conséquence une spécialisation de notre proposition, qui est adaptée ici au protocole IPv6 et à la technologie ATM.

#### 5.4.1 IPv6 et RSVP Switching

Nous proposons d'appliquer l'approche RSVP Switching dans un environnement IPv6. Cette nouvelle version du protocole IP intègre le multicast en natif ; seule l'adjonction des éléments RSVP est nécessaire pour obtenir un service multipoint avec QoS. La nouvelle version intègre un en-tête de taille fixe avec des extensions chaînées, permettant un traitement efficace du paquet. Par contre, l'accès aux champs des couches supérieures est difficile à cause de ce chaînage et d'autres obstacles tels que le cryptage des paquets. Nous présentons l'utilisation des champs IPv6 telle que nous l'envisageons dans RSVP Switching.

Un nouveau champ a été défini dans l'en-tête IPv6 : le `Flow label` (voir la Figure 2.3 de la section 2.2.3). Celui-ci nous permet d'identifier des sous-flots sans recourir à la violation de la structure en couche comme cela est nécessaire avec IPv4. La description suivante présente l'utilisation des champs IPv6 :

- L'identification de la session RSVP est réalisée avec les champs `Destination address` et `Flow label`.
- L'identification de la source pour le filtrage est réalisée avec le champ `Source address`.
- L'identification des sous-flots<sup>1</sup> peut être faite avec une partie du champ `Flow label` que nous appelons "sub" (réduisant ainsi la taille de `Flow label` pour l'identification de la session).

Une possibilité d'utilisation des champs IPv6 pour RSVP Switching est la suivante :

**Filtrage** (Table des labels dans le nœud 2) :

- "b1" port(3)out  $\mapsto$  @dest(S), Flow(S), sub1
- "b2" port(3)out  $\mapsto$  @dest(S), Flow(S), sub2

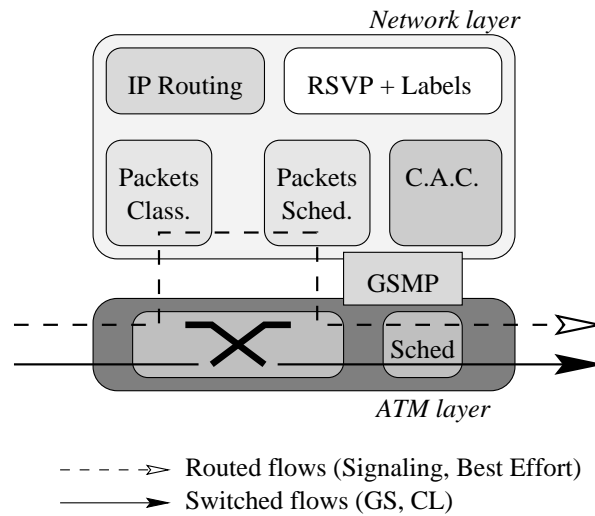
**Agrégation** (Table des labels dans le nœud 2) :

- "b1" port(3)out  $\mapsto$  @dest(S), Flow(S), @src(E1)
- "b2" port(3)out  $\mapsto$  @dest(S), Flow(S), @src(E2)

Cette mise en correspondance associe les labels de chaque port d'un commutateur avec les champs précités. "@dest" est l'adresse destination, "Flow" est le *flow label* associé à la session RSVP, "sub1" et "sub2" sont les champs utilisés pour distinguer les sous-flots et "@src" est l'adresse source.

<sup>1</sup>Précédemment, l'identification des sous-flots pouvait être réalisée par le champ `Priority` (priorités relatives à la source). Ce champ de l'en-tête ipv6 est maintenant remplacé par le champ `Class`, dans le but d'assurer une différenciation de service (priorités relatives au réseau) et ne peut donc plus être utilisé pour la mise en œuvre de nos mécanismes.





**Fig. 5.11** Architecture d'un RSVP Switch

La table des labels doit contenir les quatre champs suivants pour permettre la classification complète des paquets avec RSVP Switching : adresse destination, *flow label*, adresse source et identificateur de sous-flots.

#### 5.4.2 ATM et RSVP Switching

Dans notre proposition, nous utilisons la technologie ATM au niveau commutation. Nous nous intéressons à la partie matérielle des commutateurs ATM, qui comprend principalement la matrice de commutation et des mécanismes d'ordonnancement. Le choix d'ATM repose sur le fait que c'est la principale technologie de commutation disponible. De nombreuses autres solutions proposant de la commutation (réseaux locaux commutés) intègre aussi de la technologie ATM.

Avec ATM, les labels prennent naturellement place dans les champs VPI/VCI de l'en-tête des cellules. La table d'échange des labels au niveau de la couche de liaison correspond à la table de commutation et l'utilisation des *shortcuts* est similaire à celles des circuits virtuels ATM.

De la même manière qu'IP Switching, nous gérons directement le matériel ATM, mais sans avoir besoin d'un protocole spécifique, tel que IFMP, pour assurer l'échange des VPI/VCI. La diffusion des VPI/VCI est réalisée à travers le nouvel objet LABEL.

La Figure 5.11 présente l'architecture d'un RSVP Switch. C'est l'intégration des architectures RSVP et IP Switching. Au niveau de la couche réseau, nous trouvons les éléments classiques liés à RSVP avec une extension pour la gestion des labels dans le module RSVP. Celle-ci utilise l'objet LABEL et communique avec le commutateur ATM à travers le protocole GSMP. Au niveau de la couche liaison se trouve la partie

commutation ATM standard, où les flots routés utilisent des VC dédiés vers la partie routage du RSVP Switch, et où les flots commutés traversent seulement la matrice de commutation.

La partie ATM nécessite des capacités d'ordonnement, avec un flot par file d'attente (*per flow queuing*) ou avec un groupe de flots par file d'attente (*per class queuing*). La première technique permet de gérer simplement la QoS sur la base d'un flot (une file d'attente par VPI/VCI). La seconde est utilisée pour gérer la QoS des agrégations de flots (où plusieurs flots sont associés à la même réservation et plusieurs VPI/VCI à une file d'attente). Dans ce cas, on parle d'ordonnement par classes ; des ensembles de labels sont associés à ces classes qui représentent la réservation de la session. Cette fonctionnalité apparaît dans les commutateurs récents. Une présentation de l'ordonnement par classes se trouve dans [FJ95]. Cette technique permet d'éviter l'utilisation de commutateurs avec la fonctionnalité de *VC Merge* (commutateurs travaillant au niveau des trames pour réaliser le service N-vers-1)<sup>2</sup>.

Notre proposition n'est pas destinée au support de tous les types d'applications. Nous visons les applications avec des contraintes temporelles fortes ; ce qui demande le maintien d'un état par flot dans le réseau, que nous réalisons avec le *Soft State* de RSVP. Un des problèmes majeurs rencontrés avec ATM est la limitation de l'espace des VPI/VCI, car nous associons un identifiant VPI/VCI par flot ou sous-flot. Cela ne pose pas de problèmes au niveau d'un hôte, généralement limité à quelques centaines d'identifiants, ou à un commutateur de réseau local qui adressera la totalité des quelques 65.000 VPI/VCI. Par contre on ne peut envisager d'utiliser RSVP Switching au sein d'un réseau fédérateur, où le nombre d'identifiants serait trop faible et où la charge d'un traitement de la QoS par flots beaucoup trop élevée.

D'autres solutions, telles que celles mises en œuvre dans le groupe de l'IETF Diff-Serv (*Differentiated Services*) proposent une approche sans état bien plus légère. Mais ces solutions ne peuvent procurer que des classes de services et, en aucun cas, assurer des contraintes temporelles dures. L'approche RSVP Switching n'est pas adaptée à un environnement global, mais s'intègre plutôt à un environnement de réseaux locaux, voire de fédération de réseaux locaux à étendue limitée (réseau de campus) et aux VPN *Virtual Private Network*. L'utilisation combinée de réseaux locaux avec RSVP et de réseaux étendus est un problème important qui suscite de nombreuses recherches actuellement.

---

<sup>2</sup>Le *VC Merge* est la solution proposée par plusieurs adaptations de RSVP à la commutation. Cette solution n'est qu'une adaptation directe des mécanismes de la couche réseau. On y perd les avantages liés à la commutation de cellules (temps de latence si on rassemble les trames) et on ne peut ensuite identifier les flots multiplexés.

## 5.5 COMPARAISON AVEC D'AUTRES INTÉGRATIONS

L'apparition de la Commutation par Label en 1996 et le grand intérêt qu'elle a suscité ont entraîné plusieurs groupes de travail vers le développement d'extension du modèle de base. Dès le début de l'année 1997 plusieurs solutions pour l'intégration de RSVP ont été développées parallèlement à la notre. Dans cette section nous présentons ces différentes solutions intégrant de la QoS dans le contexte de la Commutation par Label.

### 5.5.1 IS-IPSO et QoS

IS-IPSO [AAOS98] est aussi une solution basée sur l'approche "orientée flot". Elle s'appuie sur la proposition IPSOFACTO (voir la section 3.4.1) et propose l'intégration de RSVP et des IIS (*Internet Integrated Services*) pour fournir le service IP multicast avec QoS. Ce modèle suit une approche très similaire à la notre, mais se restreint aux limitations de la technologie ATM, alors que nous proposons de résoudre les problèmes liés à ces limitations et de réaliser complètement les fonctionnalités RSVP au niveau 2.

Nous proposons dans notre solution, contrairement à la solution IS-IPSO, une méthode pour dépasser le modèle multipoint de la commutation ATM et gérer les communications multipoint-à-multipoint avec QoS proposées par RSVP, avec, en particulier, intégration des mécanismes d'agrégation de celui-ci.

### 5.5.2 MPLS et QoS

La proposition MPLS (voir la section 3.4.2) est issue de deux solutions, qui ont chacune envisagé l'intégration de RSVP. Pour le Tag Switching (voir la section 3.4.2), le document "*Tag Switching with RSVP*" [BR96] présente la manière d'associer les tags aux flots RSVP et explique comment distribuer ces tags dans la signalisation RSVP. C'est l'une des premières propositions à suggérer une extension des messages RSVP pour transporter des labels. Le groupe ARIS (voir la section 3.4.2) a proposé le document "*Soft State Switching*" [VSB97]. La réunion des travaux des deux groupes nous fournit la proposition générique présentée par MPLS : "*Use of Label Switching With RSVP*" [DRV<sup>+</sup>98].

Les propositions précédentes reposent sur l'approche "orientée signalisation" et intègrent le trafic RSVP et sa signalisation, de la même manière que les trafics IP unicast et multicast. L'attribution des labels se fait sur la base d'un label par flot RSVP<sup>3</sup>. Le transport des labels utilise la possibilité de *piggybacking* sur la signalisation RSVP, voire de l'intégration directe au protocole RSVP des labels, avec un nouvel objet défini dans RSVP pour leur transport.

---

<sup>3</sup>Un flot RSVP est identifié par un émetteur (une adresse source et optionnellement un port) et une session (un protocole, une adresse destination et optionnellement un port).

Le principe d'intégration de RSVP dans MPLS est semblable à celui d'une approche "orientée flot" mais en se plaçant dans le contexte "orienté topologie". Notre solution est plus "naturelle" par son origine "orientée flot" native liée à IP Switching.

### 5.5.3 QSR et QoS

Notre énumération ne saurait être complète sans la proposition QSR (*QoS capable Switch-Router*) [BBD<sup>+</sup>97]. Celle-ci propose la définition et l'implémentation d'un modèle focalisé sur l'intégration des flots RSVP dans un environnement IP sur ATM. La base de l'approche QSR est de commuter les flots avec QoS, le reste du trafic étant routé de manière standard. Les concepts d'extension du protocole RSVP pour le transport des labels et l'utilisation de celui-ci pour le maintien des chemins commutés sont déjà présents.

La solution proposée repose sur une implémentation spécifique de RSVP sur ATM et ne constitue pas une solution basée sur la Commutation par Label. Mais le résultat se rapproche des solutions développées pour l'intégration de RSVP. La proposition QSR développe particulièrement l'architecture matérielle et la gestion de ressources. Notre solution est plus générale et propose un service multipoint étendu.

## 5.6 MODIFICATION DE RSVP

L'utilisation d'un protocole de signalisation pour la QoS, s'est porté dans notre solution sur RSVP. Le choix de ce protocole est motivé par son adaptation à la distribution d'informations supplémentaires pour les flots. RSVP étant un protocole soutenu par l'IETF, expérimenté, et dont les codes sources de certaines implémentations sont disponibles, il nous sert de base. La modification apportée consiste en la définition d'un nouvel objet RSVP pour le support des labels. La partie contrôle de trafic et le traitement des messages doivent être étendus (voir la section 5.6.1).

Cependant les mécanismes RSVP sont conservés de même que la compatibilité avec la signalisation RSVP standard. Ceci permettra une totale interopérabilité avec une implémentation RSVP classique, en particulier à l'interface entre les zones ATM et Ethernet.

### 5.6.1 Ajout de la gestion des labels dans RSVP

Nous limitons dans un premier temps notre modification à la spécification RSVP de base définie dans le RFC 2205 [BZB<sup>+</sup>97]. Les concepts de sous-flots vus dans le modèle théorique ne sont pas introduits. Nous allons donc associer nos labels à des flots RSVP, identifiés par leurs émetteurs dans une session.

L'objet introduit respecte le format défini par la spécification de RSVP, minimisant ainsi les modifications à réaliser au traitement des messages. L'objet LABEL est pré-



Fig. 5.12 Objet LABEL

senté dans la Figure 5.12. Des valeurs libres ont été choisies pour l'identifier (**Class** = 40). Nous définissons initialement le type label seul sur 32 bits (**C-type** = 1). L'objet ainsi défini fait 8 octets de long. Il peut contenir un label ATM basé sur le VPI/VCI ou celui d'une autre technologie. L'intégration de l'objet est réalisée en le liant à un objet qui identifie le flot de données. Les labels étant nécessaires seulement en cas de réservation, nous les insérons dans la structure du message RESV qui est défini ainsi :

```

<Resv Message> ::=
    <Common Header> [ <INTEGRITY> ]
    <SESSION> <RSVP_HOP> <TIME_VALUES>
    [ <RESV_CONFIRM> ] [ <SCOPE> ]
    [ <POLICY_DATA> ... ]
    <STYLE> <flow descriptor list>

<flow descriptor list> ::=
    <empty> |
    <flow descriptor list> <flow descriptor>

```

La définition des <flow descriptor list> est spécialisée selon les différents styles disponibles :

#### WF Style :

```

<flow descriptor list> ::=
    <WF flow descriptor>

<WF flow descriptor> ::=
    <FLOWSPEC>

```

#### FF style :

```

<flow descriptor list> ::=
    <FLOWSPEC> <FILTER_SPEC> |
    <flow descriptor list> <FF flow descriptor>

<FF flow descriptor> ::=
    [ <FLOWSPEC> ] <FILTER_SPEC>

```

#### SE style :

```

<flow descriptor list> ::=
    <SE flow descriptor>

<SE flow descriptor> ::=
    <FLOWSPEC> <filter spec list>

```

```

<filter spec list> ::=
    <FILTER_SPEC> |
    <filter spec list> <FILTER_SPEC>

```

Les descripteurs de flots des styles contiennent l'information sur l'émetteur du flot dans l'objet `FILTER_SPEC`. C'est associé à celui-ci que nous localisons l'objet `LABEL`. Le style `WF` ne contient pas de `FILTER_SPEC`, nous ne l'utiliserons pas dans un premier temps. L'alternative aurait été d'associer le label à l'objet `FLOWSPEC` qui est présent dans tous les styles, mais il aurait été nécessaire d'agréger les labels dans le style `SE` et de trouver un mécanisme pour identifier les émetteurs avec le style `WF`. Les styles supportés sont définis ainsi :

#### FF style :

```

<flow descriptor list> ::=
    <FLOWSPEC> <FILTER_SPEC> [ <LABEL> ] |
    <flow descriptor list> <FF flow descriptor>

<FF flow descriptor> ::=
    [ <FLOWSPEC> ] <FILTER_SPEC> [ <LABEL> ]

```

#### SE style :

```

<flow descriptor list> ::=
    <SE flow descriptor>

<SE flow descriptor> ::=
    <FLOWSPEC> <filter spec list>

<filter spec list> ::=
    <FILTER_SPEC> [ <LABEL> ] |
    <filter spec list> <FILTER_SPEC> [ <LABEL> ]

```

Les objets `LABEL` peuvent être omis pour rester compatibles avec le trafic RSVP standard.

### 5.6.2 Souche RSVP de base

Pour notre implémentation, nous utilisons la version 4.2a3 de la distribution des sources RSVP de l'ISI (*USC Information Sciences Institute*) intégrant IPv6. Seule la gestion de la signalisation RSVP est mise en œuvre, à travers un démon situé dans l'espace utilisateur. Les fonctions associées au contrôle de trafic sont à implémenter à part. Nous avons associé un module au noyau pour l'intégration de celles-ci. C'est ce module qui contrôle effectivement l'allocation des ressources via un module d'ordonnancement des paquets à transmettre. Le démon de signalisation RSVP et le module de gestion de ressources, sont essentiels à l'obtention des services désirés et communiquent via une interface définie dans la section prochaine.

### 5.6.3 Modification de l'interface avec le noyau

Plusieurs fonctions permettent au démon RSVP, chargé de la signalisation, de dialoguer avec le module de gestion de ressources du noyau, implémenté à part. Ce sont :

```
rhandle TC_AddFlowspec(OIf, FLOWSPEC, SENDER_TSPEC,
                      ADSPEC, flags, FLOWSPEC);
FLOWSPEC TC_ModFlowspec(OIf, rhandle, FLOWSPEC, SENDER_TSPEC,
                      ADSPEC, flags, FLOWSPEC);
TC_DelFlowspec(OIf, rhandle);

fhandle TC_AddFilter(OIf, rhandle, Session, FILTER_SPEC);
TC_DelFilter(OIf, fhandle);
```

Les fonctions `TC_AddFlowspec`, `TC_ModFlowspec` et `TC_DelFlowspec` sont utilisées pour initier et modifier les paramètres de l'ordonnanceur de paquets, et les fonctions `TC_AddFilter` et `TC_DelFilter`, pour les paramètres du classifieur de paquets. Les *Flowspecs* et *Filterspecs* existant sont référencées par des pointeurs de structures (*handles*) retournés par les fonctions. Les détails de ces fonctions sont dans [BZB<sup>+</sup>97].

### Modification de l'interface

Afin de prendre en compte la gestion des labels, nous avons modifié cette interface. La liste des nouvelles fonctions relatives au contrôle de trafic est la suivante :

```
rhandle TCrsw_AddFlowspec(OIf, FLOWSPEC, SENDER_TSPEC,
                      ADSPEC, flags, FLOWSPEC);
FLOWSPEC TCrsw_ModFlowspec(OIf, rhandle, FLOWSPEC, SENDER_TSPEC,
                      ADSPEC, flags, FLOWSPEC);
TCrsw_DelFlowspec(OIf, rhandle);

fhandle TCrsw_AddFilter(OIf, rhandle, Session, FILTER_SPEC,
                      LabelSwap);
TCrsw_DelFilter(OIf, fhandle);

LABEL TCrsw_GetLabel(IIf);
TCrsw_FreeLabel(LABEL);
```

Les labels à échanger sont passés en paramètres par la fonction de contrôle de trafic `TCrsw_AddFilter`. Ces labels sont fournis à RSVP par `TCrsw_GetLabel` et rendus par `TCrsw_FreeLabel`. La description de l'implémentation des modifications de RSVP ISI se trouve dans [Die98].

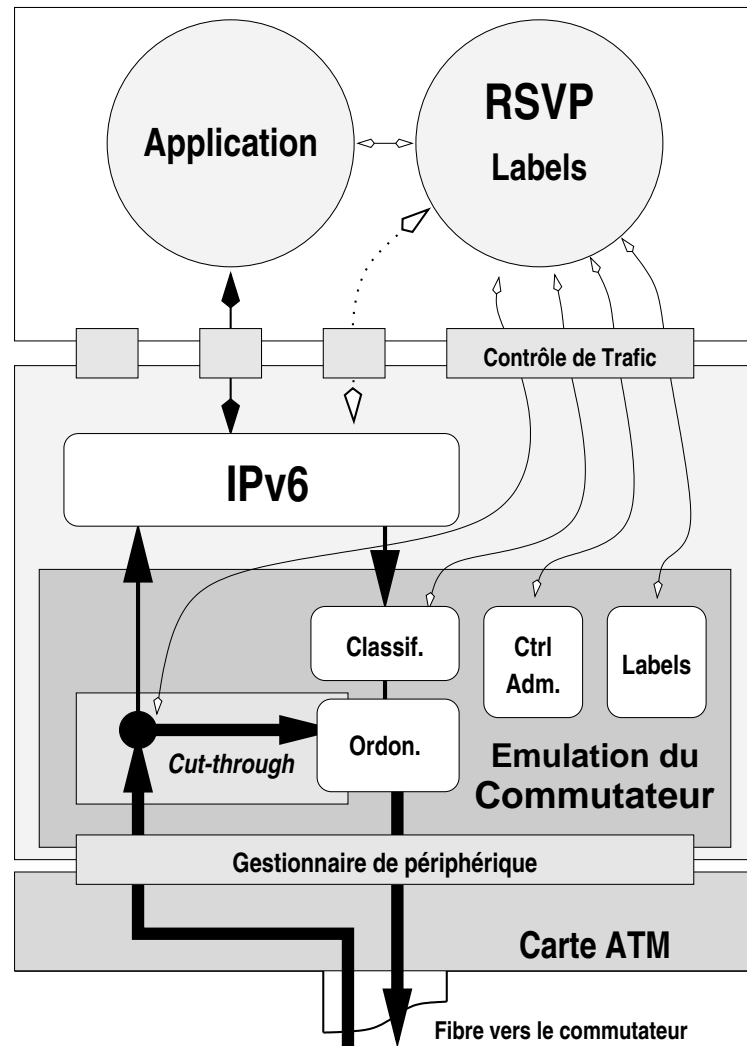


Fig. 5.13 Emulation d'un RSVP Switch

## 5.7 RÉALISATION D'UN RSVP SWITCH

La proposition RSVP Switching a pour but d'être implémentée dans un prototype, de manière à être intégrée et testée au sein de la plate-forme SAFIR (infrastructure ATM nationale succédant à MIRIHADÉ).

Notre plate-forme repose sur du matériel ATM standard, sur lequel nous ne pouvons accéder directement à la partie commutation. Il nous faut donc mettre en œuvre l'émulation d'un accès direct à cette partie du commutateur. L'émulation va être effectuée au sein de l'hôte, dans une couche intermédiaire entre le gestionnaire de la carte ATM et les modules de la couche réseau (IPv6 et RSVP).

La Figure 5.13 offre une vue d'ensemble de l'implémentation d'un RSVP Switch.



Nous distinguons de haut en bas :

- le démon RSVP, modifié afin de pouvoir gérer les labels,
- la pile IPv4/IPv6, utilisée usuellement dans le cas où aucune réservation n'est établie,
- la couche d'émulation de la commutation RSVP Switching,
- le gestionnaire de périphérique ATM.

Les trafics de données routés et commutés (trafic RSVP passant par le *cut-through*) y sont décrits. Les données commutées restent dans la couche d'émulation pour éviter le routage et simuler une commutation au niveau ATM. Les différents éléments mis en œuvre au sein de la couche d'émulation communiquent à travers l'interface du contrôle de trafic offerte par RSVP.

### 5.7.1 Plate-forme RSVP Switching

La base de notre implémentation repose principalement sur des produits issus de la communauté de recherche : une implémentation du démon RSVP présentée précédemment (voir la section 5.6.2), une souche IPv6 et un gestionnaire de carte ATM. Les machines de développement sont des PC sous GNU/Linux (voir la section 4.5.1).

#### La souche IPV6 de la DRET

La souche IPv6 utilisée est la souche DRET développée conjointement par le LIP6 et l'INRIA. Cette souche implémente un jeu réduit de fonctionnalités et a pour but de mener des expérimentations pour la gestion de la QoS. Elle repose sur un noyau Linux 2.1.23 modifié spécifiquement. Des informations sur cette souche sont disponibles à :

<http://www-rp.lip6.fr/IPv6>

#### Le gestionnaire Linux-ATM de l'EPFL

Nous utilisons les gestionnaires de carte ATM de l'EPFL (voir la section 4.5.1). Celui-ci a été modifié par l'INRIA afin de permettre un fonctionnement avec IPv6 en mode CLIP. Les connexions ATM établies sont de type PVC point-à-point, une adresse IPv6 est établie à chaque extrémité, ainsi qu'une interface logique. La table de routage IPv6 est alors mise à jour et une entrée ajoutée pour chaque PVC pour atteindre la station en regard.

### 5.7.2 Émulation d'un RSVP Switch

Nous ne pouvons ni ne souhaitons accéder directement à la partie commutation des commutateurs commerciaux dont nous disposons. Celle-ci est gérée intégralement

par le commutateur sans aucun moyen pour un hôte extérieur d'agir sur la matrice de commutation. Nous avons ainsi été conduits à proposer une architecture intégrée de RSVP Switch. Les fonctionnalités de routage et de commutation sont simultanément au sein de l'hôte.

Nous mettons en œuvre l'émulation d'un accès direct à la matrice de commutation et cette émulation va être effectuée au sein de l'hôte, dans une couche intermédiaire entre le gestionnaire de la carte ATM et les modules de la couche réseau (IPv6 et RSVP). L'émulation consiste donc à déplacer les fonctionnalités de commutation au sein de l'hôte. Ainsi la présence de réservations au sein d'un flux permet aux datagrammes associés d'être commutés dans la couche d'émulation, sans avoir besoin de remonter au niveau de la couche réseau (IPv4/6)<sup>4</sup>. Ceci est illustré sur la Figure 5.13.

Cette solution est moins performante, de par le traitement logiciel des datagrammes après réassemblage des cellules et les indirections supplémentaires, mais elle offre l'avantage d'être opérationnelle quelque soit la nature des commutateurs utilisés, et éventuellement quelque soit la nature du support de communication, ce dernier servant dans notre cas de simple lien d'interconnexion.

### Émulation de la commutation

La couche d'émulation remplace la commutation qui aurait due être effectuée dans le commutateur et le protocole de communication GSMP qui aurait dû être utilisé. La communication s'établissant directement dans l'hôte entre processus, nous n'avons plus besoin de GSMP. Les échanges entre RSVP et la nouvelle couche sont gérés principalement à travers les fonctions liées au contrôle de trafic.

La Figure 5.14 décrit les fonctions à mettre en œuvre pour l'émulation de la commutation. La couche d'émulation permet de rendre le service de Commutation par Label, attendu par la version modifiée de RSVP, tout en s'interfaçant au réseau ATM à travers le service connecté classique.

### Gestion du Multicast

Le multicast est géré au niveau de l'hôte, dans la couche d'émulation, et non dans ATM, simulant les possibilités de la matrice de commutation locale remplacée. La QoS est fournie à l'aide des mécanismes d'ordonnancement situés eux aussi dans la couche d'émulation.

---

<sup>4</sup>Seuls les datagrammes sans réservation sont traités au niveau 3. Cela inclut en particulier les messages de signalisation RSVP.

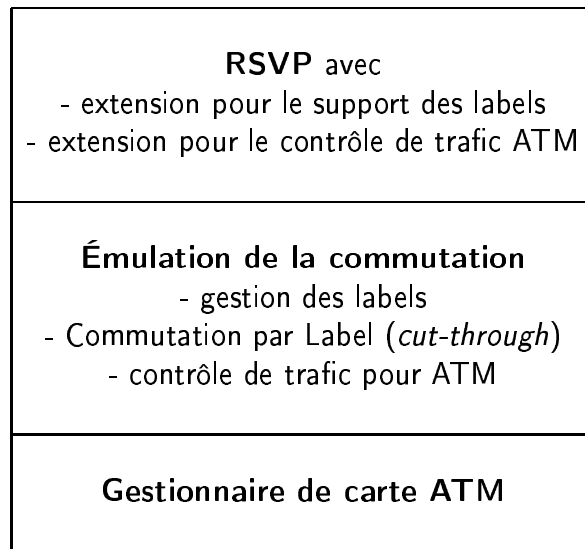


Fig. 5.14 Émulation de la commutation

### Labels utilisés

Les labels manipulés par le protocole RSVP sont des entiers sur 32 bits n'ayant pas à ce niveau de signification propre. En revanche les labels correspondent, au sein de la couche d'émulation de la commutation, au couple VPI/VCI associé au PVC. En effet, cette information, associée à l'identité de l'interface ATM de réception, permet d'identifier totalement la qualité de service associée au flux le cas échéant. De surcroît, il n'est nullement nécessaire d'inclure l'information dans les en-têtes protocolaires (IPv4/6) puisque celle-ci est associée au datagramme par le gestionnaire ATM (`struct sk_buff`).

### Table de commutation

La structure associée à chaque interface logique (une interface par PVC créé dans notre cas) possède une table de commutation. Celle-ci est indexée sur le label d'entrée et fait correspondre une interface de sortie et un label de sortie.

### Mécanismes d'ordonnement

La QoS est fournie à l'aide des mécanismes d'ordonnement situés dans la couche d'émulation au lieu d'être dans le commutateur ATM. Les datagrammes sont ensuite envoyés au sein d'un PVC UBR standard. Il n'y a pas besoin de garanties de QoS ATM au sein de ce PVC puisque ces garanties sont fournies par l'ordonnement effectué au sein de chaque hôte (voir la section 5.7.3).

## Communication avec le noyau

Les communications entre les composants du niveau utilisateur tels que le démon RSVP et les modules du noyau sont effectuées par le biais d'une nouvelle famille de protocoles (`AF_RSVP`) créée pour cet usage. Une application se contente alors d'ouvrir une *socket* en spécifiant cette famille, puis une fois le descripteur de fichier récupéré, de générer des `ioctl`s. Ces derniers sont de plusieurs types et permettent d'implémenter les fonctions de contrôle de trafic `TCrsw_*` mentionnées à la section 5.6.3.

### 5.7.3 Intégration de Service et RSVP Switching

Le groupe de travail IETF IntServ a défini deux classes de services, CL (*Controlled Load*) et GS (*Guaranteed Services*) pour l'intégration de la QoS dans l'Internet (voir la section 2.2.4). Nous présentons les mécanismes d'ordonnancement mis en œuvre.

#### Caractérisation des flots

Dans l'architecture IntServ, chaque flot est caractérisé par un TSPEC (*traffic specification*) qui comprend les cinq paramètres décrits dans le Tableau 5.2.

Paramètre	Description
$b$	Taille maximum de rafale.
$r$	Débit moyen.
$p$	Débit crête.
$L$ (ou $M$ )	Taille maximum de paquet.
$m$	Plus petite unité de poliçage.

Tab. 5.2 Les paramètres d'un TSPEC

Les trois premiers paramètres permettent de caractériser le flot selon un modèle constitué d'un **seau à jetons** (*token bucket*) qui contrôle le débit moyen et la taille de rafale, suivi d'un **seau percé** (*leaky bucket*) qui contrôle le débit crête. Les deux autres paramètres caractérisent le flot par rapport à l'implémentation des mécanismes de qualité de service. Les ressources réservées dans le réseau dépendent, pour une qualité de service donnée, de cette caractérisation.

Dans le cas du *Controlled Load*, le service fourni peut s'apparenter à une garantie de bande passante moyenne, contrairement au *Guaranteed Service* qui permet de réaliser des garanties dures et basées sur des outils mathématiques complexes [Cru96, Bou96].

### Étude du *Guaranteed Service*

La classe de service GS est conçue pour fournir aux applications une garantie stricte sur le délai d'un paquet d'un flot. Pour fournir ce type de garantie, il faut isoler complètement le flot des autres en lui fournissant un canal de communication dédié à l'intérieur de chaque lien qu'il traverse. L'émulation parfaite de ce canal est appelée le **modèle fluide**. Comme l'on se situe dans un réseau à commutation de paquets, les nœuds ne peuvent fournir qu'une approximation du service. Dans chaque nœud, la déviation entre le modèle fluide et cette approximation peut se caractériser par deux termes d'erreur que l'on définit de la manière suivante :

**C** est le terme d'erreur dépendant du débit (délai de réassemblage par exemple),

**D** est le terme d'erreur indépendant du débit (délai lié aux files d'attentes).

L'enveloppe fluide d'un flot caractérisé par le TSPEC  $(b, r, p)$  et ayant  $L$  comme taille maximum de paquet est [Ga96] :

$$A(\tau) = \min\{p.\tau, b + r.\tau\}$$

Mais il faut tenir compte du fait que le flot est constitué de paquets. Quel que soit le lissage mis en œuvre, un paquet est toujours émis au débit du lien :

$$A(\tau) = \min\{L + p.\tau, b + r.\tau\}$$

L'enveloppe du service fourni par le réseau est, dans le pire des cas, un service égal à la réservation effectuée  $R$  après un délai  $\sum(\frac{C_i}{R} + D_i)$  sans service (avec  $C_i$  et  $D_i$  les termes C et D du nœud  $i$ ).

On peut ainsi calculer le délai maximum subi par un paquet avec  $n$  le nombre de nœuds :

$$D = \frac{(b-L)(p-R)}{R(p-r)} + \frac{L}{R} + \sum_{i=1}^n \left[ \frac{C_i}{R} + D_i \right] \quad (5.1)$$

A partir de l'équation précédente, on peut obtenir la réservation nécessaire pour un délai donné :

$$R = \frac{p.x + \sum_{i=1}^n C_i}{x + D - \sum_{i=1}^n D_j} \quad \text{avec} \quad x = \frac{b-L}{p-r} \quad (5.2)$$

### Proposition de service *Controlled Load*

L'implémentation du *Guaranteed Service* résulte en une faible utilisation de la bande passante du lien, puisque la réservation  $R$  est toujours plus grande que le débit moyen

du flot. Cette bande passante n'est pas perdue et reste disponible pour le trafic *Best Effort* si on utilise un ordonnanceur qui conserve le travail tel que WFQ (*Weighted Fair Queuing*) [DKS89].

Cependant, cette bande passante ayant été réservée, elle peut être utilisée pour fournir des garanties de services moins contraignantes que le GS à d'autres flots. Nous proposons donc d'assurer une classe de service permettant d'utiliser ces réservations pour fournir une garantie de bande passante moyenne. L'idée de ce service est similaire à ce qui est présenté dans [Ga96] et correspond aux besoins de la classe de service CL. Nous proposons donc à travers ce service appelé BR (*Bandwidth Recovery*) une implémentation possible de la classe CL, conçue pour être implémentée conjointement avec la classe GS de façon à optimiser l'utilisation des ressources.

Pour implémenter la classe BR, nous utilisons un ordonnanceur WFQ avec des files séparées pour chaque flot GS ou BR. À chaque flot GS est affecté un poids qui détermine la quantité de bande passante  $R_i$  allouée à ce flot. Pour  $N$  flots GS, chacun caractérisé par une enveloppe  $(b, r, p)$  et un lien de capacité  $C_l$ , garantir le service aux flots GS implique le respect de l'inégalité suivante :

$$\sum_{i=1}^N R_i \leq C_l - \Delta_{BR} - \Delta_{IP} \quad (5.3)$$

Dans l'équation précédente,  $\Delta_{BR}$  et  $\Delta_{IP}$  représentent les portions de bande passante réservées respectivement aux flots BR et trafic sans QoS. Le poids associé au flot sans QoS est basé sur une estimation de la bande passante nécessaire au trafic restant. Le poids associé à chaque flot BR est choisi très petit. Il permet aux flots BR d'accéder à la bande passante réservée mais pas utilisée par les flots GS. Considérons  $N'$  flots BR avec un débit moyen égal à  $r'$ , leurs garanties de débit minimum à long terme est possible si :

$$\sum_{i=1}^N r_i + \sum_{i=1}^{N'} r'_i \leq C_l - \Delta_{IP} \quad (5.4)$$

L'équation (5.4) assure que si les flots GS respectent leur contrat de trafic, les flots BR recevront chacun un débit minimum garanti  $r'_i$  sur une échelle de temps raisonnable.

### L'ordonnanceur du RSVP Switch

Dans un RSVP Switch, les flots avec QoS sont commutés au niveau ATM. L'ordonnanceur mettant en œuvre les mécanismes décrits ci-dessus pour l'implémentation des services GS et BR est donc situé au niveau ATM. Au niveau routage, on fait la distinction entre deux types de trafic : celui de signalisation et le trafic *Best Effort*. Le trafic de signalisation étant prioritaire sur le trafic *Best Effort*, l'ordonnanceur

au niveau routage est un FIFO à priorités avec deux files. Le trafic du niveau IP est placé dans une file de priorité inférieure de façon à ne pas perturber les flots demandant une QoS.

Avec l'introduction de la couche d'émulation, la gestion de l'ordonnancement se retrouve au niveau de l'hôte. Nous intégrons ainsi les ordonnanceurs du niveau routage et commutation. L'utilisation d'un seul WFQ est envisageable et nous pouvons utiliser à la place du terme  $\Delta_{IP}$  les termes  $\Delta_{BE}$  et  $\Delta_{SIG}$  pour séparer les réservations des flots de la signalisation et du trafic *Best Effort*.

## 5.8 L'EXPÉRIMENTATION

Nous utilisons comme support d'expérimentation de notre proposition RSVP Switching le projet DGA/MENERT appelé DIS/ATM auquel nous participons avec le LAAS, l'INRIA Sophia Antipolis et Dassault Électronique. À travers ce projet nous mettons en œuvre une architecture fournissant les nouvelles fonctionnalités dans le cadre des réseaux ATM. Cette architecture repose sur une infrastructure basée sur plusieurs RSVP Switch assurant un service réseau performant avec multipoint et QoS.

### 5.8.1 Les applications DIS

Le projet DIS/ATM consiste en l'expérimentation d'une série d'applications, utilisant le protocole DIS (*Distributed Interactive Simulation*), fournies par Dassault Électronique [Dul97] dans un environnement IPv6 sur réseau ATM.

La Figure 5.15 présente les différentes fenêtres liées à l'application. Sur cet exemple mettant en œuvre deux hélicoptères (correspondant à deux applications DIS actives), on peut observer les panneaux de contrôle des hélicoptères (**SSTH Helico**), un écran de visualisation du scénario (**SSTH Animation**) avec une représentation en deux dimensions du champ de bataille et des trajectoires des deux hélicoptères. Un tableau d'analyse statistique des trafics associés au DIS (**SSTH Analyse**) avec une sous-fenêtre indiquant le débit instantané d'un des trafics surveillés est également visualisé.

Les fenêtres sont ici regroupées, mais le but est évidemment de distribuer les applications sur un grand nombre d'hôtes et de sites. La version complète du logiciel intègre d'autres types de visualisation, et en particulier, un module en trois dimensions ajoutant au réalisme de la simulation.





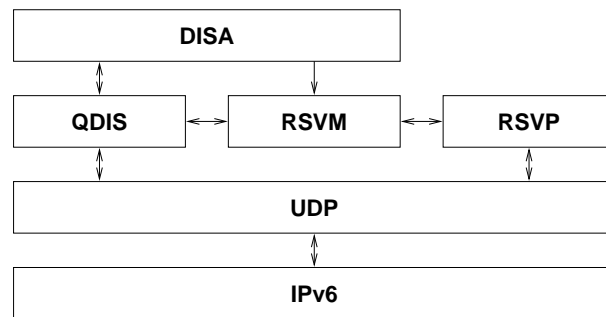


Fig. 5.16 Architecture DIS avec QoS

### 5.8.2 Intégration du DIS avec RSVP

Les applications DIS ne sont pas prévues initialement pour spécifier leurs besoins en terme de QoS. Des travaux ont été réalisés au sein du LAAS pour caractériser le trafic DIS et proposer une architecture étendue avec gestion de la QoS.

L'architecture proposée par le LAAS intègre un module intermédiaire permettant d'interfacer les applications DIS avec RSVP [CDD<sup>+</sup>98]. La Figure 5.16 présente ce module, appelé RSVM (*Ressource reSerVation Manager*). Celui-ci communique directement avec les applications DIS (DISA), le protocole DIS agrémenté de la QoS (QDIS) et RSVP. Le tout repose sur le service UDP/IPv6.

Notre travail supporte donc l'architecture précédente en fournissant un service IPv6 avec RSVP dans un environnement ATM.

### 5.8.3 Intégration du DIS avec RSVP Switching

L'utilisation de notre solution RSVP Switching sur la plate-forme SAFIR permet la réalisation efficace d'un service multipoint avec QoS sur un réseau haut débit et longue distance. La configuration de la plate-forme ATM pour les expérimentations envisagées est présentée sur la Figure 5.17. Sur ce schéma, seules les connexions ATM sont représentées pour les stations disposant d'une interface ATM directe. Toutes les stations sont aussi rattachées à travers des réseaux locaux classiques (*Ethernet*) à l'Internet. Nous pouvons ainsi réaliser un grand nombre de configurations protocolaires afin de comparer les différentes solutions envisagées :

- Influence de la distance, grâce à des topologies locales ou longue distance.
- Comparaison de RSVP Switching par rapport à RSVP + IP/Ethernet.
- Comparaison de RSVP Switching par rapport à RSVP + IP/LANE.

Des stations communiquent indirectement avec la plate-forme en utilisant RSVP non modifié sur réseau classique afin de démontrer la compatibilité et l'interfonctionnement avec les protocoles existants. D'autres trafics que ceux des applications

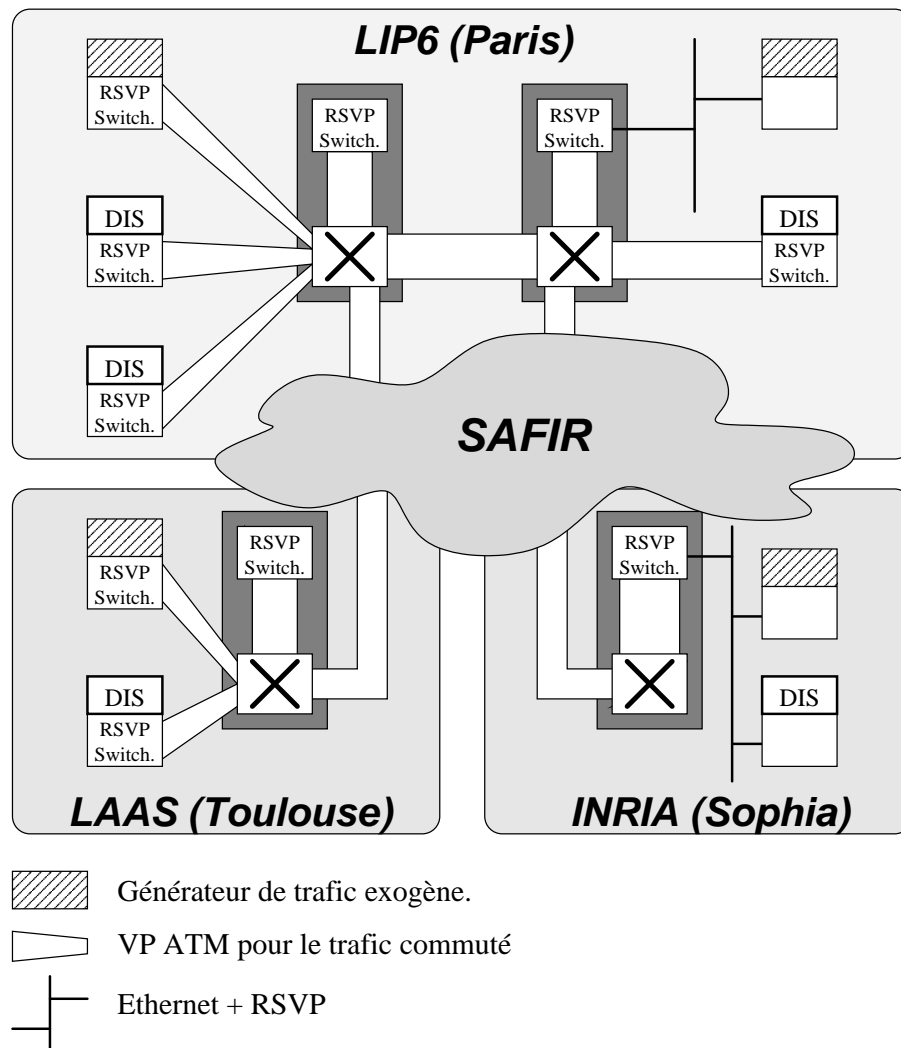


Fig. 5.17 Configuration de la plate-forme ATM

DIS sont prévus parmi les trafics gérés par les RSVP Switch afin de mesurer leur influence et le respect des garanties de QoS.

#### 5.8.4 Observations

L'intérêt de l'approche RSVP Switching réside dans la possibilité d'utiliser les performances de la commutation ATM sans être limité par le mode connecté standard. Notre approche nous offre ainsi la possibilité d'utiliser le faible temps de latence de la commutation de cellules et les débits élevés, tout en utilisant directement la richesse fonctionnelle de la couche IP avec par exemple RSVP. Notre approche nous permet d'envisager une intégration efficace de services multipoints avec QoS sur les réseaux à haute performance basés sur la commutation. L'implémentation de notre

approche à travers les RSVP Switches est en cours de finalisation. La mise en œuvre de la plate-forme expérimentale et l'obtention des premiers résultats sont prévues à des dates avoisinant celle de la soutenance de cette thèse.

## 5.9 CONCLUSION DU CHAPITRE

Nous proposons, à travers RSVP Switching, une solution permettant les nouvelles fonctionnalités avec des performances élevées. Notre approche trouve son application directe dans l'environnement IP sur ATM. Elle permet l'intégration de la commutation ATM au niveau liaison, avec IPv6 pour le niveau réseau. Cela nous permet d'élaborer une solution pour fournir le modèle de communication multipoint-à-multipoint proposé par RSVP dans une approche basée sur la commutation IP. Les matériels ATM récents proposent des capacités de gestion locale de la QoS importantes, notamment au niveau de la mise en œuvre de files d'attente par flot. Les limitations de notre architecture vont plutôt se situer au niveau de la partie contrôle où la gestion d'un nombre important d'états est difficile.

Notre solution est plus particulièrement adaptée aux environnements locaux (réseaux de campus ou VPN) à cause des limitations de RSVP en terme d'adaptation aux grands groupes. L'approche RSVP Switching est dédiée aux environnements où les garanties de QoS sont plus importantes que le nombre de flots. La standardisation, au sein du groupe MPLS, d'un protocole de signalisation pour la commutation (LDP) complété de mécanismes de gestion de la QoS est en cours. Cette alternative permettra peut-être d'accéder à des solutions pour l'intégration des nouvelles fonctionnalités aux environnements commutés moins sensibles au facteur d'échelle. Il se posera alors la question de la mise en correspondance des services liés à LDP avec ceux de RSVP, ce dernier se développant particulièrement au niveau des interfaces applicatives.

Pour valider expérimentalement notre solution nous l'implémentons sur notre plate-forme ATM. Les limitations d'accès à la commutation ATM nous amènent à proposer un modèle émulant la commutation. Notre prototype a pour but de supporter une application de DIS avec de fortes contraintes en terme de QoS.

## CHAPITRE 6

# *Conclusion*

L'objectif de cette thèse est l'étude de l'intégration du multipoint et de la QoS dans les réseaux haut débit. Pour atteindre ce but, nous associons une signalisation liée aux nouvelles fonctionnalités avec une signalisation pour les hautes performances. Pour la première signalisation, nous choisissons le protocole RSVP et son service multipoint-à-multipoint avec QoS hétérogène et dynamique. La signalisation haut débit est, quant à elle, envisagée à travers deux signalisations accompagnant la commutation, l'une basée sur mode connecté et l'autre non. Notre contribution s'appuie donc sur l'utilisation conjuguée de RSVP et des réseaux basés sur la commutation, et plus particulièrement ceux basés sur la technique de commutation ATM.

RSVP et la commutation ATM sont associés à deux technologies de réseaux très prometteuses dans leurs domaines respectifs : IP pour RSVP, qui se présente comme la couche de convergence *de facto* pour l'accès aux services réseaux, et ATM qui est une base des futures infrastructures de réseaux haut débit multimédia. Leurs champs d'action étant en partie complémentaires, ces deux technologies sont amenées à coopérer. C'est pourquoi nous avons étudié les communications multipoints avec QoS à travers l'intégration d'IP sur ATM. Cette intégration est complexe du fait des modes de fonctionnement orthogonaux des deux technologies utilisées.

Plusieurs approches sont envisageables pour réaliser cette intégration. Dans un premier temps nous avons considéré une approche traditionnelle, basée sur la signalisation usuelle en mode connecté des réseaux ATM. L'approche choisie se base sur CLIP, solution expérimentée et présente dans la plupart des équipements ATM. Cependant, le service fourni ne supporte ni le multipoint, ni la QoS. Les dévelop-

pements récents ajoutent le multipoint à travers MARS, et nous introduisons le service avec QoS grâce à une intégration classique de RSVP. Nous proposons une solution répondant au service attendu par le modèle RSVP. L'utilisation de multiples arbres multipoints pour transporter différents paliers de QoS permet d'assurer l'hétérogénéité des récepteurs tout en tirant parti des possibilités des connexions ATM point-à-multipoint.

Même si notre solution pallie certaines limitations des connexions ATM, telles que l'homogénéité et l'absence de renégociation de la QoS, les limitations qui sont apparues à travers nos expérimentations sont liées à un autre aspect du mode orienté connexion et de sa signalisation associée. Le service point-à-multipoint des réseaux ATM a un temps de latence pour la création d'une connexion qui croît linéairement en fonction du nombre de récepteurs en plus du nombre d'éléments intermédiaire. Cette caractéristique rend impropre la technologie ATM associée à une signalisation orientée connexion au support du service multipoint que nous voulons fournir, celui-ci se devant de traiter un grand nombre de participants dynamiquement.

Dans un deuxième temps, nous envisageons une nouvelle solution, basée aussi sur la commutation, mais avec une signalisation non connectée. Notre approche, appelée "RSVP Switching", repose sur le protocole RSVP associé au concept de "Commutation par Label". À l'aide d'une extension, nous intégrons la signalisation de commutation au protocole RSVP. Cela permet une gestion des flots commutés par la signalisation RSVP au niveau routage, tout en offrant la performance de la commutation aux flots avec QoS. La possibilité d'utiliser le faible temps de latence et les débits élevés de la commutation, tout en profitant directement de la signalisation RSVP, nous laisse à penser qu'une intégration efficace du multipoint avec QoS est possible. Appliquée à la commutation ATM, notre approche permet d'éviter l'utilisation de la signalisation orientée connexion associée aux réseaux ATM. Les limitations de notre architecture se trouvent au niveau de la partie contrôle où la gestion de l'état du réseau par RSVP n'est peut-être pas adaptée aux grands groupes. Notre solution est donc à réserver aux environnements réduits à cause de ces limites. L'approche RSVP Switching est dédiée aux environnements où les garanties de QoS sont plus importantes que le nombre de flots.

Nous choisissons de valider expérimentalement notre solution à travers une implémentation sur notre plate-forme ATM longue distance. Les limitations d'accès à la partie commutation des équipements ATM nous amènent à proposer un modèle émulant la commutation au niveau d'un hôte directement connecté au réseau ATM. Notre prototype a pour but de répondre aux besoins d'un service IPv6/RSVP/IntServ supporté efficacement par la commutation ATM. Ce service étant requis par des applications de DIS avec des contraintes importantes de QoS. L'expérimentation a lieu sur une plate-forme ATM longue distance reliant différents sites équipés d'hôtes directement rattachés au réseau ATM et intégrant RSVP Switching.

La contribution de ce travail est de proposer une solution pour l'intégration des services multipoints avec QoS dans les réseaux haut débit. Nous prôtons, outre l'utilisation du mode non connecté pour la signalisation de commutation, une signalisation intégrée. Celle-ci vise l'utilisation associée de la signalisation pour les nouvelles fonctionnalités et de la signalisation haut débit. Notre solution prend en compte le multipoint avec QoS alors que la tendance du domaine serait plutôt à miser uniquement sur le gain en performance, ce qui revient à négliger la facette multimédia des besoins des applications. Notre solution apporte aussi une réponse efficace au problème de l'adaptation d'IP sur ATM pour les services multipoints avec QoS de type RSVP.

### Perspectives

Le sujet de notre contribution est lié à des domaines de recherches très actifs tels que les réseaux haut débit, les réseaux multimédia ou les signalisations pour la commutation. Ce dernier domaine est récent et son actualité nous place dans un environnement à forte concurrence ; des contributions intéressantes sont souvent élaborées parallèlement. Notre approche visant l'intégration de RSVP a aussi intéressé d'autres groupes, confirmant par là les choix effectués.

L'intégration des nouvelles fonctionnalités avec la commutation étant plus que jamais d'actualité, nous pensons continuer dans cette voie. Notre réflexion ne portera pas forcément sur RSVP car son association avec la commutation peut être considérée comme une intégration de celle-ci dans RSVP. D'autres approches sont envisageables, telles que l'intégration directe du multipoint et de la QoS dans la signalisation de commutation. Au sein de MPLS, d'importantes discussions ont eu lieu pour savoir si la QoS devait être intégrée à LDP. L'extension de LDP vers les nouvelles fonctionnalités fait partie des travaux dans la continuité de notre approche. D'autres problématiques importantes liées à la QoS peuvent être incorporées. Citons notamment le routage en fonction de la QoS (*QoS routing*) et le routage explicite, souvent lié à RSVP.



# Publications

- [1] O. Fourmaux. “Gestion de Flots de Données Multicast, Dynamiques, et Hétérogènes avec RSVP sur les Réseaux ATM”, *Actes des Journées Doctorales en Informatique et Réseau (JDIR'96)*, Paris, France, septembre 1996.
- [2] O. Fourmaux. “Adaptation du Protocole RSVP sur les Réseaux ATM”, *Rapport de recherche IBP-MASI 96/24*, novembre 1996, 40p.
- [3] S. Fdida, O. Fourmaux & R. Onvural. “Enabling Multimedia Networks”, *Electronic Journal on Networks and Distributed Processing (EJNDP)*, ISSN 1262-3261, vol 5, march 1997, pp. 28-32.
- [4] O. Fourmaux & C. Deleuze. “Qualité de service hétérogène et dynamique pour liaisons ATM point-à-multipoint”, *Actes de la Conférence Francophone sur l'Ingénierie des Protocoles (CFIP'97)*, Liège, Belgique, septembre 1997.
- [5] O. Fourmaux. “Suivi de l'intégration d'IP sur ATM”, *Contrat DCN Ingénierie, Toulon, rapport initial en janvier 1997, rapports complémentaires en mai, septembre et décembre 1997, rapport final en janvier 1998*.
- [6] O. Fourmaux. “Suivi des standards ATM Forum : MPOA”, *Contrat DCN Ingénierie, Toulon, rapport initial en janvier 1997, rapports complémentaires en mai, septembre et décembre 1997, rapport final en janvier 1998*.
- [7] O. Fourmaux & E. Anique. “QoS Issues for Multicasting”, *COST 237 Final Report*, A. Danthine editor. À paraître chez Springer-Verlag, 1998.
- [8] O. Fourmaux, V. Roca, C. Deleuze & S. Fdida. “Architecture réseau pour le projet DIS/ATM : RSVP Switching, une solution pour un service multipoint avec QoS en environnement IPv6 sur ATM”, *Rapport pour le projet DIS/ATM avec le LAAS, l'INRIA et Dassault Electronique*, septembre 1998.
- [9] O. Fourmaux & S. Fdida. “Multicast for RSVP Switching – An Extended Multicast Model with QoS for Label Swapping in an IP over ATM Environment”, À paraître dans *Telecommunication Systems*, 1998.





# Bibliographie

- [AAOS98] Arup Acharya, Furquan Ansari, Max Ott, and Henning Sanneck. Dynamic QoS for IP switching using RSVP over IPSOFACTO. In *Proceedings of SYBEN '98*, Zurich, May 1998.
- [ADA97] Arup Acharya, Rajiv Dighe, and Furquan Ansari. IP Switching Over Fast ATM Cell Transport (IPSOFACTO): Switching multicast flows. In *Proceedings of IEEE GLOBECOM '97*, volume 3, pages 1850–1854, Austin, Texas, November 1997.
- [ADF<sup>+</sup>98] L. Andersson, P. Doolan, N. Feldman, A. Fredette, and R. Thomas. Label Distribution Protocol (LDP) Specification. Internet Draft <draft-ietf-ldp-spec-00.txt>, work in progress, IETF, March 1998.
- [Alm96] Werner Almesberger. ATM on Linux. Technical report, Laboratoire de Réseaux de Communication (LRC), École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland, March 1996.
- [Arm96] G. Armitage. Support for Multicast over UNI 3.0/3.1 based ATM Networks. IETF Request For Comment 2022, Bellcore, November 1996.
- [Arm97a] Grenville Armitage. IP Multicasting over ATM Networks. *IEEE Journal on Selected Areas in Communications*, 15(3):445–457, April 1997.
- [Arm97b] Grenville Armitage. VENUS : Very Extensive Non-Unicast Service. Internet Draft <>, work in progress, February 1997.
- [ASJH98a] Grenville Armitage, Peter Schuler, Markus Jork, and Geraldine Harter. IPV6 over Non-Broadcast Multiple Access (NBMA) Networks. Internet Draft <draft-ietf-ion-ipv6-01.txt>, work in progress, IETF, March 1998.
- [ASJH98b] Grenville Armitage, Peter Schuler, Markus Jork, and Geraldine Harter. IPv6 over ATM Networks. Internet Draft <draft-ietf-ion-ipv6-atm-01.txt>, work in progress, IETF, March 1998.

- 
- [Atk94] R. Atkinson. Default IP MTU for use over ATM AAL5. Request For Comment 1626, IETF, May 1994.
- [BBD<sup>+</sup>97] E. Basturk, A. Birman, G. Delp, R. Guérin, et al. Design and Implementation of QoS Capable Switch-Router. Technical Report RC 20848 (01/31/97) Computer Science, IBM Research Division, January 1997.
- [BCDB95] M. Borden, E. Crawley, B. Davie, and S. Batsell. Integration of Real-time Services in an IP-ATM Network Architecture. Informational Request For Comment 1821, IETF, August 1995.
- [BCS94] R. Braden, D. Clark, and D. Shenker. Integrated Services in the Internet Architecture: an Overview. Informational Request For Comment 1633, IETF, July 1994.
- [Ber98a] L. Berger. RSVP over ATM Implementation Guidelines. Request For Comment 2379, IETF, August 1998.
- [Ber98b] L. Berger. RSVP over ATM Implementation Requirements. Request For Comment 2380, IETF, August 1998.
- [BFGK95] A. Birman, V. Firoiu, R. Guérin, and D. Kandlur. Provisioning of rsvp-based services over a large atm network. Technical Report RC 20250 (10/27/95) Computer Science, IBM Research Division, 1995.
- [Bou96] Jean-Yves Le Boudec. Network Calculus Made Easy. Technical Report EPFL-DI 96/128, Laboratoire de Réseaux de Communication (LRC), École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland, December 1996.
- [BR96] F. Baker and Y. Rekhter. Tag Switching with RSVP. Internet Draft <draft-baker-tag-rsvp-00.txt>, work in progress, IETF, December 1996.
- [BZB<sup>+</sup>97] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jasmin. Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification. Request For Comment 2205, IETF, September 1997.
- [Cal97] Julien Calm. ELVViS! : Une Application de Vidéoconférence avec Codage Hiérarchique et Multicast. Rapport de stage, DEA Système Informatiques, Université Pierre et Marie Curie, August 1997. devenu LaViD.
- [Can98] D. Cansever. NHRP Protocol Applicability Statement. Request For Comment 2333, IETF, April 1998.
- [Cas92] S. Casner. First IETF Internet Audiocast. *ACM Computer Communication Review*, 22(3):92–97, July 1992.

- 
- [CBB<sup>+</sup>98] E. Crawley, L. Berger, S. Berson, F. Baker, M. Borden, and J. Krawczyk. A Framework for Integrated Services and RSVP over ATM. Informational Request For Comment 2382, IETF, August 1998.
- [CCI91] B-ISDN Protocol Reference Model and its Application. Recommendation I.321, CCITT, April 1991.
- [CD95] A. Conta and S. Deering. Internet Control Message Protocol (ICMPv6 for the internet protocol version 6 (ipv6). Request For Comment 1885, IETF, December 1995.
- [CDD<sup>+</sup>98] C. Chassot, L. Dairaine, M. Diaz, A. Lozes, and L. Rojas. Réserveation de ressources pour applications DIS en environnement Internet nouvelle génération. Technical report, LAAS-CNRS, 1998.
- [CDF<sup>+</sup>97] R. Callon, P. Doolan, N. Feldman, A. Fredette, G. Swallow, and A. Viswanathan. A Framework for MultiProtocol Label Switching. Internet Draft <draft-ietf-mpls-framework-02.txt>, work in progress, IETF, November 1997.
- [Ciz98] G. Cizault. *IPv6 : théorie et pratique*. O'Reilly, March 1998.
- [CPS94] Jean-Pierre Coudreuse, Gérard Pays, and Helga Seguin. ATM : le mode de transfert asynchrone – principes, applications et perspectives. Mémento technique 4, France Telecom / Conseil Scientifique, October 1994. Les communications à large bande.
- [Cru96] R. Cruz. Quality of Service Guarantees in Virtual Circuit Switched Networks. *IEEE Journal on Selected Areas in Communications*, 13(6):1048–1056, August 1996.
- [CSV96] R. Cole, D. Shur, and C. Villamizar. IP over ATM: A Framework Document. Request For Comment 1932, IETF, April 1996.
- [DC84] S. Deering and D. Cheriton. Host groups: A multicast extension for datagram internetworks. In *Proceedings of ACM SIGCOMM '84*, volume 14, September 1984.
- [DDC97] Christophe Diot, Walid Dabbous, and Jon Crowcroft. Multipoint Communication: A Survey of Protocols, Functions and Mechanisms. *IEEE Journal on Selected Areas in Communications*, 15(3):277–290, April 1997.
- [DDK<sup>+</sup>97] P. Doolan, B. Davie, D. Katz, Y. Rekhter, and E. Rosen. Tag Distribution Protocol (TDP) Specification. Internet Draft <draft-doolan-tdp-spec-01.txt>, work in progress, IETF, May 1997.

- 
- [DDL<sup>+</sup>97] B. Davie, P. Doolan, J. Lawrence, K. McCloghrie, Y. Rekhter, E. Rosen, and G. Swallow. Use of Tag Switching With ATM. Internet Draft <draft-davie-tag-atm-01.txt>, work in progress, IETF, January 1997.
- [DDR98] Bruce Davie, Paul Doolan, and Yakov Rekhter. *Switching in IP Networks: IP Switching, Tag Switching, and related Technologies*. Number ISBN 1-55860-505-3 in Networking. Morgan Kaufmann, 340 Pine Street, San Francisco, CA 94104, May 1998.
- [Dee89] S. Deering. Host Extensions for IP Multicasting. IETF Request For Comment 1112, Stanford University, August 1989.
- [Dee91] S. Deering. *Multicast Routing in Datagram Internetwork*. PhD thesis, Stanford University, December 1991.
- [DH95] S. Deering and R. Hinden. Internet Protocol Version 6 (IPv6) Specification. Request For Comment 1883, IETF, December 1995.
- [Die98] Tomas Dietz. Label Swapping with RSVP in an IP over ATM Environment. Diplomarbeit am Institut für Telematik, Fakultät für Informatik - Universität Karlsruhe, May 1998.
- [DKS89] Alan Demers, Srinivasan Keshav, and Scott Shenker. Analysis and Simulation of a Fair Queueing Algorithm. In *Proceedings of ACM SIGCOMM '89*, volume 19, Austin, Texas, USA, September 1989. ACM Computer Communication Review.
- [DLM<sup>+</sup>97] B. Davie, J. Lawrence, K. McCloghrie, Y. Rekhter, E. Rosen, G. Swallow, and P. Doolan. Use of Label Switching With ATM. Internet Draft <draft-davie-mpls-atm-00.txt>, work in progress, IETF, November 1997.
- [dP95] Martin de Prycker. *Asynchronous Transfer Mode - Solution for broadband ISDN*. Hellis Horwood, third edition, 1995.
- [DRV<sup>+</sup>98] B. Davie, Y. Rekhter, A. Viswanathan, S. Blake., V. Srinivasan, and E. Rosen. Use of Label Switching With RSVP. Internet Draft <draft-ietf-mpls-rsvp-00.txt>, work in progress, IETF, March 1998.
- [Dul97] Didier Dulac. Spécification de l'application DIS. Rapport provisoir de projet, Dassault Electronique, May 1997.
- [EKM<sup>+</sup>96] H. Esaki, Y. Katsube, S. Matsuzawa, A. Mogi, K. Nagami, and T. Jinmei. Cell switch router: High performance packet forwarding architecture over internet and intranet using atm technology. White paper, Toshiba Corporation, Japan, November 1996.

- 
- [Eri94] H. Eriksson. MBone: The Multicast Backbone. *Communication of the ACM*, 37:54–60, August 1994.
- [FD97] Olivier Fourmaux and Christophe Deleuze. Qualité de service hétérogène pour liaisons ATM point-à-multipoint. In *Actes de CFIP'97*, pages 383–398, Liège, Belgium, September 1997. Université de Liège, Hermes.
- [FDFA98] N. Feldman, P. Doolan, A. Fredette, and L. Andersson. MultiProtocol Label Switching Architecture. Internet Draft <draft-ietf-mpls-arch-01.txt>, work in progress, IETF, March 1998.
- [FF98] Olivier Fourmaux and Serge Fdida. Multicast for RSVP Switching – An Extended Multicast Model with QoS for Label Swapping in an IP over ATM Environment. *Telecommunication Systems*, 1998. To be published.
- [FFO97] Serge Fdida, Olivier Fourmaux, and Raif Onvural. Enabling multimedia networks. *Electronic Journal on Networks and Distributed Processing (EJNDP)*, ISSN 1262-3261, 5:28–32, March 1997.
- [FJ95] Sally Floyd and Van Jacobson. Link-Sharing and Ressource Management Models for Packet Networks. *IEEE/ACM Transactions on Networking*, 3(4):365–386, August 1995.
- [FLYV93] V. Fuller, T. Li, J. Yu, and K. Varadhan. Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy. IETF Request For Comment 1519, IETF, September 1993.
- [FM96] Norman Finn and Tony Mason. ATM LAN Emulation. *IEEE Communications Magazine*, 34(6):96–100, June 1996.
- [Fou96] Olivier Fourmaux. Gestion de Flots de Données Multicast, Dynamiques et Hétérogènes avec RSVP sur les Réseaux ATM. In *Actes de JDIR'96*, pages 1–15–1–20, Paris, France, September 1996. École Nationale Supérieure des Télécommunications.
- [FRDF98] Olivier Fourmaux, Vincent Roca, Cristofe Deleuze, and Serge Fdida. Architecture réseau pour le projet DIS/ATM : RSVP Switching, une solution pour un service multipoint avec QoS en environnement IPv6 sur ATM. Technical report, LIP6, September 1998.
- [FV97] N. Feldman and A. Viswanathan. ARIS Specification. Internet Draft <draft-feldman-aris-spec-00.txt>, work in progress, IBM Corporation, March 1997.

- 
- [Ga96] L. Georgiadis and al. Efficient support of delay and rate guarantees in an internet. In *Proceedings of ACM SIGCOMM '96*, volume 26, Palo Alto, California, USA, September 1996. ACM Computer Communication Review.
- [GaJBS97] R. Gilligan, S. Thomson and J. Bound, and W. Stevens. Basic Socket Interface Extensions for IPv6. Informational Request For Comment 2133, IETF, January 1997.
- [GB98] Mark W. Garrett and Marty Borden. Interoperation of Controlled-Load Service and Guaranteed Service with ATM. Request For Comment 2381, IETF, August 1998.
- [GBO97] Eric Gauthier, Jean-Yves Le Boudec, and Philippe Oechslin. SMART: A many-to-many multicast protocol for atm. *IEEE Journal on Selected Areas in Communications*, 15(3):458–471, April 1997.
- [GR97] M. Grossglauser and K. K. Ramakrishnan. SEAM: Scalable and efficient atm multipoint-to-multipoint multicasting. In *Proceedings of IEEE INFOCOM '97* [INF97].
- [Gué96] R. Guérin. Qos support in an internet – definitions, issues, and challenges. Cost 242 closing seminar, IBM T.J. Watson Research Center, June 1996.
- [Hei93] J. Heinanen. Multiprotocol Encapsulation over ATM Adaptation Layer 5. IETF Request For Comment 1483, Telecom Finland, July 1993.
- [Hui98] Christian Huitéma. *IPv6, The New Internet Protocol*. Prentice Hall, second edition, 1998.
- [ILM96] Integrated Local management Interface Specification – Version 4.0. Technical Report af-ilmi-00065.000, ATM Forum, September 1996.
- [INF97] *IEEE INFOCOM'97*, number 16, Kobe, Japan, April 1997.
- [ITU95] User-Network Interface (UNI) Layer 3 Specification for Basic Call/Connection Control. Recommendation Q.2931, ITU-T, 1995.
- [KKK<sup>+</sup>98] W. Knitl, A. Kujooory, B. Klessig, R. Dantu, P. Robinson, B. Petri, J. Drake, and H. Suzuki. QOS Extensions to MPOA. LANE-MPOA Subworking Group Contribution 98-0299, ATM-Forum, April 1998.
- [KNE97] Y. Katsube, K. Nagami, and H. Esaki. Toshiba's Router Architecture Extensions for ATM: Overview. IETF Informational Request For Comment 2098, Toshiba R&D Center, February 1997.

- 
- [KON97] Y. Katsube, Y. Ohba, and K. Nagami. Two Modes of MPLS Explicit Label Distribution Protocol. Internet Draft <draft-katsube-mpls-two-ldp-00.txt>, work in progress, Toshiba, September 1997.
- [Kum96] V. Kumar. *MBone: Interactive Multimedia on the Internet*. New Riders, Indianapolis, 1996.
- [LAN95] Lan emulation over atm – version 1.0 specification. Technical Report af-lane-00021.000, ATM Forum, January 1995.
- [LAN97a] Lan emulation over atm – version 2.0, lnni specification. Technical Report Draft 8, btd-lane-lnni-02.08, ATM Forum, February 1997.
- [LAN97b] Lan emulation over atm – version 2.0, luni specification. Technical Report af-lane-00084.000, ATM Forum, July 1997.
- [Lau93] M. Laubach. Classical IP and ARP over ATM. IETF Request For Comment 1577, Hewlett-Packard Laboratories, December 1993.
- [LH98] M. Laubach and J. Halpern. Classical IP and ARP over ATM. IETF Request For Comment 2225, Com21, Inc. and Newbridge Networks, Inc., April 1998. *Remplace les RFC 1577 et 1626*.
- [LKP<sup>+</sup>98] J. Luciani, D. Katz, D. Piscitello, B. Cole, and N. Doraswamy. NBMA Next Hop Resolution Protocol (NHRP). Request For Comment 2332, IETF, April 1998.
- [LM97] Steven Lin and Nick McKeown. A Simulation Study of IP Switching. In *Proceedings of ACM SIGCOMM '97*, volume 27, Cannes, France, September 1997. ACM Computer Communication Review.
- [LMH<sup>+</sup>97] F. Liaw, A. Mankin, E. Hoffman, D. Grossman, and A. Malis. ATM Signaling Support for IP over ATM. Request For Comment 1755, IETF, February 1997.
- [Mah98] M. Maher. ATM Signalling Support for IP over ATM - UNI Signalling 4.0 Update. Request For Comment 2331, IETF, April 1998.
- [MPO97] Multiprotocol over atm – version 1.0. Technical Report af-mpoa-00087.000, ATM Forum, July 1997.
- [NBM<sup>+</sup>97] Douglas Niehaus, Abdella Battou, Andrew McFarland, Basil Decina, Henry Dardy, Vinai Sirkay, and Bill Edwards. Performance Benchmarking of Signaling in ATM Networks. *IEEE Communications Magazine*, pages 134–143, August 1997.



- [NEH<sup>+</sup>96a] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall. Ipsilon Flow Management Protocol Specification for IPv4 – Version 1.0. IETF Informational Request For Comment 1953, Ipsilon, May 1996.
- [NEH<sup>+</sup>96b] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall. Ipsilon’s General Switch Management Protocol Specification – Version 1.1. IETF Informational Request For Comment 1987, Ipsilon, August 1996.
- [NEH<sup>+</sup>96c] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall. Transmission of Flow Labelled IPv4 on ATM Data Links – Version 1.0. IETF Informational Request For Comment 1954, Ipsilon, May 1996.
- [NEH<sup>+</sup>98] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall. Ipsilon’s General Switch Management Protocol Specification Version 2.0. IETF Informational Request For Comment 2297, Nokia, Sprint, Fiberlane, March 1998.
- [NKS<sup>+</sup>97] K. Nagami, Y. Katsube, Y. Shobatake, A. Mogi S. Matsuzawa, T. Jinmei, and H. Esaki. Toshiba’s Flow Attribute Notification Protocol (FANP) Specification. IETF Informational Request For Comment 2129, Toshiba R&D Center, April 1997.
- [NLM96] Peter Newman, Tom Lyon, and Greg Minshall. Flow Labelled IP: A Connectionless Approach to ATM. In *Proceedings of IEEE INFOCOM ’96*, volume 3, pages 1251–1260, San Francisco, California, USA, March 1996.
- [NML97] Peter Newman, Greg Minshall, and Tom Lyon. IP Switching: ATM Under IP. *IEEE/ACM Transactions on Networking*, November 1997.
- [NMLH97] Peter Newman, Greg Minshall, Tom Lyon, and Larry Huston. IP Switching and Gigabit Routers. *IEEE Communications Magazine*, April 1997.
- [NNS96] T. Narten, E. Nordmark, and W. Simpson. Neighbor Discovery for Internet Protocol Version 6 (IPv6). Request For Comment 1970, IETF, August 1996.
- [OEN94] M. Otha, H. Esaki, and K. Nagami. Conventional IP over ATM. Internet Draft <draft-otha-ip-over-atm-00.txt>, work in progress, IETF, March 1994.
- [Oos97] Simon Oosthoek. Survey of Multicast Support for IP over ATM, with an implementation in mind. Master’s thesis, University of Twente, 1997.

- 
- [ORC<sup>+</sup>98] P. Owezarski, V. Roca, P. Cipièere, S. Fdida, M. Diaz, and C. Diot. Protocoles multimédia haut débit et expérimentations sur MIRIHADÉ. Technical report, LAAS-CNRS, LIP6-CNRS, INRIA Sophia-Antipolis, 1998.
- [OS96] R. O. Onvural and V. Srinivasan. A Framework for Supporting RSVP Flows Over ATM Networks. Internet Draft <draft-onvural-rsvp-atm-00.txt>, February 1996.
- [Pe81] Jon Postel and editor. Internet Protocol, darpa internet program protocol specification. Request For Comment 791, IETF, September 1981.
- [RDK<sup>+</sup>97] Y. Rekhter, B. Davie, D. Katz, E. Rosen, and G. Swallow. Cisco Systems' Tag Switching Architecture Overview. IETF Informational Request For Comment 2105, Cisco Systems Inc., February 1997.
- [RDS<sup>+</sup>97] Y. Rekhter, B. Davie, G. Swallow, D. Farinacci, and D. Katz. Tag switching architecture overview. In *Proceedings of the IEEE INFOCOM '97* [INF97], pages 1973–1983.
- [SCS<sup>+</sup>97] L. Salgarelli, A. Corghi, M. Smirnow, H. Sanneck, and D. Witaszek. Supporting IP Multicast Integrated Services in ATM Networks. Internet Draft <draft-salgarelli-iss11-mis-00.ps>, work in progress, CEFRIEL Politecnico di Milano and GMD-FOKUS, November 1997.
- [Smi97] M. Smirnow. Scalable and efficient multiprotocol ip multicast over atm. In *Proceedings of SPIE VV'97*, Dallas, Texas, nov 1997. SPIE.
- [SPG97] S. Shenker, C. Partridge, and R. Guerin. Specification of Guaranteed Quality of Service. Request For Comment 2212, IETF, September 1997.
- [SRC84] J. Saltzer, D. Reed, and D. Clark. End-to-End Arguments in System Design. *ACM Transactions on Computers Systems*, 2(4):277–288, 1984.
- [TA97] R. Talpade and M. Ammar. Multicast Server Architectures for MARS-based ATM multicasting. IETF Informational Request For Comment 2149, Georgia Institute of Technology, May 1997.
- [TM96] Traffic Management Specification – Version 4.0. Technical Report af-tm-00056.000, ATM Forum, April 1996.
- [UNI93] *ATM User-Network Interface Specification – Version 3.0*. Prentice Hall, September 1993. ISBN 0-13-225863-3.
- [UNI94] *ATM User-Network Interface Specification – Version 3.1*. Prentice Hall, September 1994. ISBN 0-13-393828-X.
- [UNI96] ATM User-Network Interface (UNI) Signaling Specification – Version 4.0. Technical Report af-sig-00061.000, ATM Forum, July 1996.

- [VFBW97] A. Viswanathan, N. Feldman, R. Boivie, and R. Woundy. ARIS: Aggregate Route-based IP Switching. Internet Draft <draft-viswanathan-aris-overview-00.txt>, work in progress, IBM Corporation, March 1997.
- [VSB97] A. Viswanathan, V. Srinivasan, and S. Blake. Soft State Switching – A Proposal to Extend RSVP for Switching RSVP Flows. Internet Draft <draft-viswanathan-mpls-rsvp-00.txt>, work in progress, IBM Corporation, August 1997.
- [Wro97a] J. Wroclawski. Specification of the Controlled-Load Network Element Service. IETF Request For Comment 2211, MIT LCS, September 1997.
- [Wro97b] J. Wroclawski. The Use of RSVP with IETF Integrated Services. IETF Request For Comment 2210, MIT LCS, September 1997.
- [ZDE<sup>+</sup>93] Lixia Zhang, Stephen Deering, Deborah Estrin, Scott Shenker, and Daniel Zappala. RSVP: A New Resource ReSerVation Protocol. *IEEE Network Magazine*, pages 8–18, September 1993.

# *Liste des acronymes*

AAL	<i>ATM Adaptation Layer</i>
ABR	<i>Available Bit Rate</i>
API	<i>Application Programming Interface</i>
ARIS	<i>Aggregate Route-based IP Switching</i>
ATM	<i>Asynchronous Transfer Mode</i>
ATMARP	<i>ATM Address Resolution Protocol</i>
ARP	<i>Address Resolution Protocol</i>
BGP	<i>Border Gateway Protocol</i>
B-ISDN	<i>Broadband Integrated Services Digital Network</i>
B-ICI	<i>B-ISDN InterCarrier Interface</i>
BR	<i>Bandwidth Recovery</i>
BUS	<i>Broadcast and Unknown Server</i>
CBR	<i>Contituous Bit Rate</i>
CBT	<i>Core Based Tree</i>
CCITT	Comité Consultatif International Télégraphique et Téléphonique
CIDR	<i>Classless Internet Domain Routing</i>
CL	<i>Controlled-Load</i>
CLIP	<i>Classical IP over ATM</i>
CLP	<i>Cell Loss Priority</i>
CNET	Centre National d'Etudes des Télécommunications
CNRS	Centre National de la Recherche Scientifique
COLIP	<i>COventional IP over ATM</i>
CPCS	<i>Common Part Convergence Sublayer</i>
CSR	<i>Cell Switch Router</i>
DIS	<i>Distributed Interactive Simulation</i>
DVMRP	<i>Distance Vector Multicast Routing Protocol</i>

ELAN	<i>Emulated Local Area Network</i>
EPFL	École Polytechnique Fédérale de Lausanne
EARTH	<i>EAsy IP multicast Routing THrough ATM</i>
FANP	<i>Flow Attribute Notification Protocol</i>
FDDI	<i>Fiber-Distributed Data Interface</i>
FEC	<i>Forwarding Equivalence Classes</i>
FIB	<i>Forwarding Information Base</i>
GFC	<i>Generic Flow Control</i>
GNU	<i>GNU is Not Unix</i>
GPL	<i>GNU General Public Licence</i>
GSMP	<i>General Switch Management Protocol</i>
GS	<i>Guaranteed Service</i>
HEC	<i>Header Error Control</i>
IASG	<i>Internetwork Address Sub-Group</i>
ICMP	<i>Internet Control Message Protocol</i>
IETF	<i>Internet Engineering Task Force</i>
IFMP	<i>Ipsilon Flow Management Protocol</i>
IGMP	<i>Internet Group Management Protocol</i>
IIS	<i>Internet Integrated Services</i>
IISSL	<i>Internet Integrated Services over Specific Layer</i>
ILMI	<i>Integrated Local Management Interface</i>
InARP	<i>Inverse Address Resolution Protocol</i>
INRIA	Institut National de Recherche en Informatique et en Automatique
ION	<i>Internetworking Over NBMA</i>
IP	<i>Internet Protocol</i>
IPATM	<i>IP Over ATM</i>
IRC	<i>Internet Relay Chat</i>
IRTF	<i>Internet Research Task Force</i>
IS	<i>Integrated Services</i>
ISDN	<i>Integrated Services Digital Network</i>
ISI	<i>Information Sciences Institute</i>
ISO	<i>International Organization for Standardization</i>
ITU-T	<i>International Telecommunication Union – Telecommunication</i>

---

LAN	<i>Local Area Network</i>
LANE	<i>Local Area Network Emulation</i>
LDP	<i>Label Distribution Protocol</i>
LEC	<i>LAN Emulation Client</i>
LECS	<i>LAN Emulation Configuration Server</i>
LES	<i>LAN Emulation Server</i>
LIJ	<i>Leaf Initiated Join</i>
LIP6	Laboratoire d'Informatique de PARIS VI
LIS	<i>Logical IP Subnetwork</i>
LLC	<i>Logical Link Control</i>
LNNI	<i>LAN Emulation Network-Network Interface</i>
LSR	<i>Label Switch Router</i>
LSP	<i>Label Switched Path</i>
LUNI	<i>LAN Emulation User-Network Interface</i>
MAC	<i>Medium Access Control</i>
MAN	<i>Metropolitan Area Network</i>
MARS	<i>Multicast Address Resolution Server</i>
MBONE	<i>Multicast backBONE</i>
MCS	<i>MultiCast Server</i>
MIS	<i>Multicast Integration Server</i>
MLIS	<i>Multicast Logical IP Subnet</i>
MOSPF	<i>Multicast Open Shortest Path First</i>
MPC	<i>MultiProtocol Client</i>
MPLS	<i>MultiProtocol Label Switching</i>
MPOA	<i>MultiProtocol Over ATM</i>
MPS	<i>MultiProtocol Server</i>
MTU	<i>Maximum Transmission Unit</i>
NBMA	<i>NonBroadcast Multiple Access</i>
NHC	<i>Next Hop Client</i>
NHRP	<i>Next Hop Resolution Protocol</i>
NHS	<i>Next Hop Server</i>
nrt-VBR	<i>Non Real-Time Variable Bit Rate</i>
OPWA	<i>One Path With Advertising</i>
OSI	<i>Open Systems Interconnection</i>

OSPF	<i>Open Shortest Path First</i>
PDU	<i>Protocol Data Unit</i>
PIM	<i>Protocol Independent Multicast</i>
PIM-DM	<i>Protocol Independent Multicast – Dense Mode</i>
PIM-SM	<i>Protocol Independent Multicast – Sparse Mode</i>
PNNI	<i>Private Network to Network Interface</i>
PT	<i>Payload Type</i>
PVC	<i>Permanent Virtual Connection</i>
QoS	<i>Quality of Service</i>
QSR	<i>QoS capable Switch-Router</i>
RFC	<i>Request For Comment</i>
RIJ	<i>Root Initiated Join</i>
RNIS-LB	<i>Réseau Numérique à Intégration de Service à Large Bande</i>
ROLC	<i>Routing Over Large Clouds</i>
RPF	<i>Reverse Path Forwarding</i>
RSVP	<i>Resource ReserVation Protocol</i>
RTP	<i>Real-time Transport Protocol</i>
rt-VBR	<i>Real-Time Variable Bit Rate</i>
SAP	<i>Service Access Point</i>
SAR	<i>Segmentation And Reassembly</i>
SEAM	<i>Scalable and Efficient ATM Multipoint-to-Multipoint Multicasting</i>
SMART	<i>Shared Many-to-many ATM ReservaTion</i>
SNAP	<i>SubNetwork Access Protocol</i>
S.R.	<i>Serveur RSVP</i>
STM	<i>Synchronous Transfer Mode</i>
SVC	<i>Switched Virtual Connection</i>
TCP/IP	<i>Transmission Control Protocol/Internet Protocol</i>
TDP	<i>Tag Distribution Protocol</i>
TIB	<i>Tag Information Base</i>
TM	<i>Traffic Management</i>
TOS	<i>Type Of Service</i>
TSR	<i>Tag Switch Router</i>
TTL	<i>Time To Live</i>
UBR	<i>Unspecified bit rate</i>

---

UNI	<i>User-Network Interface</i>
UIT-T	Union Internationale des Télécommunications – secteur des Télécommunications
VBR	<i>Variable Bit Rate</i>
VCI	<i>Virtual Channel Identifier</i>
VC	<i>Virtual Channel</i>
VENUS	<i>Very Extensive Non Unicast Service</i>
VPI	<i>Virtual Path Identifier</i>
VP	<i>Virtual Path</i>
VPN	<i>Virtual Private Network</i>
WAN	<i>Wide Area Network</i>
WFQ	<i>Weighted Fair Queuing</i>
WRR	<i>Weighted Round Robin</i>
YAAAAA	<i>Yet Another Annoying And Ambiguous Acronym</i>





# Liste des figures

2.1	L'interconnexion de sous-réseaux. . . . .	10
2.2	L'abstraction de groupe avec IP multicast. . . . .	12
2.3	Comparaison des en-têtes de paquets IP. . . . .	14
2.4	Architecture des nœuds du réseau avec RSVP. . . . .	18
2.5	Structure d'une cellule ATM . . . . .	20
2.6	Principe de la commutation de cellules . . . . .	21
2.7	La pile protocolaire ATM . . . . .	22
2.8	Les différentes interfaces des réseaux ATM . . . . .	23
2.9	Structure d'une cellules ATM, à travers UNI . . . . .	23
2.10	Services de base . . . . .	24
2.11	Maillage Multicast ATM . . . . .	27
2.12	Serveur Multicast ATM . . . . .	28
3.1	Encapsulation CLIP . . . . .	34
3.2	Résolution d'adresses avec CLIP . . . . .	35
3.3	Maillage multicast avec MARS . . . . .	37
3.4	Serveur multicast avec MARS . . . . .	38
3.5	Transmission vers un groupe multi-cluster . . . . .	39
3.6	Les raccourcis de NHRP . . . . .	42
3.7	La couche LAN Emulation . . . . .	44
3.8	Les différentes connexions de LANE . . . . .	45
3.9	Communication multicast avec LANE . . . . .	46
3.10	Utilisation d'un <i>shortcut</i> MPOA . . . . .	48
3.11	Les composants MPOA . . . . .	48
3.12	Les connexions MPOA . . . . .	49
3.13	Résolution d'adresses étendue avec NHRP dans MPOA . . . . .	50
3.14	Création d'un <i>Shortcut</i> MPOA . . . . .	50
3.15	Moins de couches intermédiaires! . . . . .	52
3.16	Comparaison de l'établissement d'un shortcut . . . . .	53
3.17	L'architecture d'un IP Switch . . . . .	55
3.18	Les étapes de création d'un chemin commuté . . . . .	56
3.19	Principes du CSR . . . . .	57
3.20	Principes d'IPSOFACTO . . . . .	58
3.21	Architecture d'un modèle basé sur la topologie . . . . .	59

3.22	Diffusion des tags et transmission avec Tag Switching . . . . .	61
3.23	Établissement des circuits commutés avec ARIS . . . . .	62
3.24	Le multicast avec IP Switching . . . . .	63
3.25	Mise en place du multicast MPLS . . . . .	65
4.1	VC dédié à RSVP entre deux éléments connectés à ATM . . . . .	71
4.2	RSVP dans un hôte ou un routeur connecté à ATM . . . . .	72
4.3	Modèle de base . . . . .	73
4.4	Modèle à connectivité globale . . . . .	73
4.5	Modèle serveur avec LIS . . . . .	74
4.6	Modèle serveur en connectivité globale . . . . .	75
4.7	VC pour le style Wildcard-Filter . . . . .	76
4.8	VC pour le style Shared-Explicit . . . . .	77
4.9	VC pour le style Fixed-Filter . . . . .	77
4.10	Réservation avec une QoS unique . . . . .	81
4.11	Réservation avec QoS multiples . . . . .	83
4.12	Réservation avec plusieurs VC point-à-multipoint . . . . .	84
4.13	Réservation de VC point-à-multipoint avec flots séparés . . . . .	86
4.14	Réservation de VC point-à-multipoint avec codage hiérarchique . . . . .	87
4.15	Réservation classique à travers trois LIS . . . . .	90
4.16	Réservation d'un "shortcut" à travers trois LIS. . . . .	91
4.17	Réservation d'un "shortcut" Multicast à travers 3 LIS. . . . .	92
4.18	Gestion d'un flot par les RCE. . . . .	93
4.19	Mesures sur la plate-forme MIRIHADÉ . . . . .	98
4.20	Principe de l'expérimentation vidéo . . . . .	100
4.21	Interface de l'application LaViD . . . . .	100
4.22	Différentes qualités obtenues avec LaViD . . . . .	101
4.23	Expérimentation sur MIRIHADÉ . . . . .	102
4.24	Schéma de câblage des PVC point-à-multipoint . . . . .	103
5.1	Chemin de routage initial . . . . .	113
5.2	Création du chemin avec label . . . . .	113
5.3	Création d'un <i>cut-through</i> unicast . . . . .	114
5.4	Création d'un <i>cut-through</i> multicast . . . . .	114
5.5	Chemin initial routé pour le filtrage . . . . .	116
5.6	Association des labels pour le filtrage . . . . .	117
5.7	Création du <i>cut-through</i> pour le filtrage . . . . .	117
5.8	Chemin initial routé pour l'agrégation . . . . .	119
5.9	Association des labels pour l'agrégation . . . . .	119
5.10	Création du <i>cut-through</i> pour l'agrégation . . . . .	120
5.11	Architecture d'un RSVP Switch . . . . .	122
5.12	Objet LABEL . . . . .	126
5.13	Emulation d'un RSVP Switch . . . . .	129
5.14	Émulation de la commutation . . . . .	132

---

5.15 Applications DIS . . . . .	137
5.16 Architecture DIS avec QoS . . . . .	138
5.17 Configuration de la plate-forme ATM . . . . .	139



# Liste des tableaux

2.1	Les classes de service de l'IntServ . . . . .	16
2.2	Les fonctionnalités de RSVP . . . . .	16
2.3	Les styles RSVP . . . . .	18
2.4	Les paramètres de QoS dans les réseaux ATM . . . . .	26
2.5	Les catégories de services des réseaux ATM . . . . .	26
2.6	Comparaison des modèles de services . . . . .	30
4.1	Les types de VC supportés par le <i>driver</i> de l'EPFL . . . . .	95
4.2	Les services supportés par le <i>driver</i> de l'EPFL . . . . .	96
4.3	Temps de création des VC point-à-multipoint . . . . .	98
5.1	Comparaison des niveaux routage et commutation . . . . .	111
5.2	Les paramètres d'un TSPEC . . . . .	133

Thèse réalisée avec

**L<sup>A</sup>T<sub>E</sub>X**

sur système

**GNU/Linux**