



HAL
open science

Architectures distribuées temps réel fondées sur ATM

Jean-Francois Guillaud

► **To cite this version:**

Jean-Francois Guillaud. Architectures distribuées temps réel fondées sur ATM. Réseaux et télécommunications [cs.NI]. Institut National Polytechnique de Grenoble - INPG, 1995. Français. NNT : . tel-00005047

HAL Id: tel-00005047

<https://theses.hal.science/tel-00005047>

Submitted on 24 Feb 2004

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.

THESE

Présentée à

l'Institut National Polytechnique de Grenoble

Pour obtenir le titre de **Docteur**
(Arrêté ministériel du 30 Mars 1992)
Spécialité : Informatique

par

Jean-François GUILLAUD

Architectures distribuées temps réel fondées sur ATM

Soutenue le 19 Octobre 1995 devant la commission d'examen

Composition du jury :

Jacques MOSSIERE	<i>Président</i>
Daniel ETIEMBLE	<i>Rapporteur</i>
Marc LOBELLE	<i>Rapporteur</i>
Pascal SICARD	<i>Examineur</i>
Gérard MICHEL	<i>Directeur de thèse</i>

Thèse préparée au sein du Laboratoire de Génie Informatique - IMAG

A mes Parents

Remerciements

Je voudrais tout d'abord remercier Monsieur Jacques Mossière, Directeur du Laboratoire Logiciels, Systèmes et Réseaux (LSR) de l'IMAG, pour l'honneur qu'il me fait de présider le jury de cette thèse.

J'adresse tous mes remerciements à Messieurs Daniel Etiemble, Professeur à l'Université Paris Sud, et Marc Lobelle, Professeur à l'Université Catholique de Louvain, pour avoir accepté de rapporter sur mes travaux.

Je tiens également à remercier Pascal Sicard, Maître de Conférence à l'Université Joseph Fourier, d'avoir accepté de faire partie de ce jury.

Je remercie tout particulièrement Gérard Michel, Professeur ENSIMAG, pour m'avoir accueilli dans son équipe de recherche et m'avoir prodigué les conseils nécessaires à l'élaboration de cette thèse.

Je remercie aussi les ingénieurs d'ITMI APTOR pour leur concours dans toute la partie technique de ces travaux, et tout particulièrement Yves Ansade d'avoir accepté de me co-encadrer et ainsi assurer le financement de ces trois années de recherche, Jean-Luc Dubois pour nos entretiens constructifs lors de la réalisation des spécifications, Sébastien Bouat, Michel Aubert et Christian Arandyelovitch pour leurs conseils et la mise en oeuvre des résultats de cette thèse, et tous ceux qui, de près ou de loin, ont participé à la réussite de ces travaux.

Je tiens également à remercier Jean-Daniel Laisné, Joseph Le Drezen, Gérald Ouvradou et Bernt Weber de Télécom Bretagne qui ont mené à bien la réalisation du micro-contrôleur.

Cette page ne serait pas complète sans dire merci à tous mes collègues du Laboratoire de Génie Informatique qui ont su me faire passer ces trois années dans la bonne humeur.

"Plus on en sait et plus on en ignore"

Jean d'Ormesson

Table des Matières

Table des Matières	9
Liste des Figures	15
Liste des Tables	17
Chapitre I : Introduction	19
I.1. Les applications distribuées temps réel	22
I.2. Les limites des solutions actuelles	23
I.3. L'adéquation de la technique ATM aux applications distribuées temps réel	24
I.4. Les objectifs	25
I.5. Le plan de la thèse	26
Chapitre II : Les grands projets dans l'architecture des machines pour les hauts débits	29
II.1. La conception d'architectures pour les hauts débits.....	31
II.1.1. Les contraintes des nouvelles applications	32
II.1.2. L'implantation des protocoles de communication	33
II.1.3. Résumé	35
II.2. Les interfaces réseau	35
II.2.1. La réduction des copies et des transferts sur le bus	36
II.2.1.1. Les optimisations logicielles.....	37
II.2.1.2. Les optimisations matérielles	38
II.2.2. La répartition de l'exécution des protocoles entre la carte d'interface et la machine	40

II.2.3. L'intégration avec le système d'exploitation.....	41
II.2.4. Résumé	42
II.3. Les machines de communication	42
II.3.1. Le remplacement du bus fond de panier par un réseau	43
II.3.1.1. Desk Area Network	43
II.3.1.2. Micronet Machines	44
II.3.1.3. ATOMIC	45
II.3.2. L'extraction des flux audio/vidéo au plus près du réseau	46
II.3.2.1. VuNet et ViewStation.....	46
II.3.2.2. Washington University multimedia workstation	47
II.3.2.3. Le système de l'Université de Lancaster	48
II.3.3. L'ajout d'un bus pour le traitement audio/vidéo	49
II.3.3.1. La station multimédia de NEC	49
II.3.3.2. Le système de l'Université de Melbourne	50
II.3.4. Résumé	51
II.4. Conclusion	52
Chapitre III : L'architecture du système de communication proposé	53
III.1. L'architecture de couplage des équipements	55
III.1.1. Les caractéristiques matérielles	55
III.1.1.1. Présentation générale	55
III.1.1.1.1. Le coupleur ATM	55
III.1.1.1.2. Les topologies variées	57
III.1.1.2. Les choix de conception	60
III.1.1.2.1. L'interface physique au réseau et la commutation intégrée	61
III.1.1.2.2. La carte mère d'adaptation à ATM	63
III.1.1.2.2.1. Le choix du type d'AAL	63
III.1.1.2.2.2. La gestion du tampon de segmentation et de réassemblage ...	65
III.1.1.2.2.3. Le mode de transfert (flot ou message).....	67
III.1.1.2.2.4. La capacité des tampons	69
III.1.1.2.2.4.1. Le tampon de segmentation et de réassemblage	69
III.1.1.2.2.4.2. Le tampon d'émission et de réception	70
III.1.1.2.3. Le fonctionnement sans micro-contrôleur local	71
III.1.1.2.4. Le micro-contrôleur local.....	72

III.1.1.2.4.1. L'architecture de la carte fille micro-contrôleur local	73
III.1.1.2.4.2. Le fonctionnement avec le micro-contrôleur local	75
III.1.1.2.4.3. Les types critiques de transfert d'information	76
III.1.1.2.4.3.1. Les flux à optimiser	76
III.1.1.2.4.3.2. Le fonctionnement	78
III.1.1.2.4.3.3. Le concept.....	81
III.1.1.3. Résumé	82
III.1.2. Les caractéristiques logicielles	82
III.1.2.1. Présentation générale	83
III.1.2.2. Les choix de conception	85
III.1.2.2.1. Le couplage entre ATM et les applications	85
III.1.2.2.1.1. Les fonctions à ajouter à l'AAL pour offrir le service requis ...	85
III.1.2.2.1.2. SSCS transport rapide.....	87
III.1.2.2.1.2.1. Le protocole.....	88
III.1.2.2.1.2.2. La qualité de service	89
III.1.2.2.1.3. SSCS transferts critiques.....	89
III.1.2.2.1.4. SSCS signalisation	89
III.1.2.2.1.5. Le support des autres protocoles	90
III.1.2.2.2. Les protocoles supportés	91
III.1.2.2.2.1. La signalisation et la méta-signalisation	91
III.1.2.2.2.2. IP sur ATM.....	91
III.1.2.2.2.2.1. Le modèle IP classique	92
III.1.2.2.2.2.2. L'encapsulation des datagrammes IP sur AAL 5	93
III.1.2.2.2.2.3. La résolution d'adresses	93
III.1.2.2.2.3. Les autres protocoles	94
III.1.2.2.3. Les interfaces offertes aux applications	94
III.1.2.2.3.1. L'accès au transport rapide	94
III.1.2.2.3.2. L'accès aux transferts critiques	97
III.1.2.2.3.3. TCP et UDP	99
III.1.2.3. Résumé	99
III.1.3. Conclusion	99
III.2. La gestion de la communication entre les équipements	101
III.2.1. Le contrôle de congestion	101
III.2.1.1. Introduction.....	101
III.2.1.2. Le contrôle d'accès au réseau	102
III.2.1.3. L'évitement de congestion.....	103

III.2.1.4. La gestion des trafics	106
III.2.1.5. Résumé	107
III.2.2. Le mode non connecté	107
III.2.2.1. Introduction.....	107
III.2.2.2. IP sur ATM	108
III.2.2.3. LAN Emulation.....	109
III.2.2.4. Résumé	110
III.2.3. La signalisation	110
III.2.3.1. Introduction.....	110
III.2.3.2. La présentation du modèle	111
III.2.3.3. La signalisation	112
III.2.3.3.1. Les messages de signalisation	112
III.2.3.3.2. Le protocole de signalisation UNI	113
III.2.3.3.3. Le protocole de signalisation NNI	114
III.2.3.4. L'ouverture de connexions par une tierce partie	117
III.2.3.5. Résumé	118
III.2.4. L'administration du réseau	118
III.2.4.1. Introduction.....	118
III.2.4.2. Le modèle d'administration	119
III.2.4.3. Les particularités de l'administration de notre système	120
III.2.4.4. Résumé	121
III.2.5. La gestion de la redondance et la reconfiguration	122
III.2.5.1. Introduction.....	122
III.2.5.2. La gestion de fautes et la redondance	122
III.2.5.2.1. Introduction	122
III.2.5.2.2. Les fonctions OAM.....	122
III.2.5.2.3. La redondance	123
III.2.5.3. La reconfiguration	126
III.2.5.4. Résumé	127
III.2.6. Conclusion	127
III.3. Conclusion.....	128
Chapitre IV : L'évaluation des performances	129
IV.1. Les objectifs	131
IV.2. Le langage VHDL et la simulation des systèmes	132

IV.2.1. Présentation de VHDL	132
IV.2.2. L'utilisation de VHDL pour l'évaluation de performances	133
IV.3. Le modèle	134
IV.4. La comparaison des performances	137
IV.4.1. Les éléments à comparer	137
IV.4.2. Les performances du système pour l'envoi de messages classiques	140
IV.4.3. Les performances pour le transfert périodique de variables temps réel	147
IV.5. Conclusion	152
Chapitre V : L'étude de coût	155
V.1. Introduction	157
V.2. Le coût de la solution sans le micro-contrôleur local	158
V.3. Le coût de la solution avec le micro-contrôleur local	159
V.4. Conclusion	160
Chapitre VI : Conclusion	161
VI.1. Les résultats	163
VI.2. Les perspectives	166
Références	169
Annexe A : Glossaire des abréviations utilisées	181
Annexe B : La technique ATM	189
B.1. Les caractéristiques d'ATM	191
B.1.1. Introduction	191
B.1.2. Les principes fondamentaux d'ATM	192
B.1.3. Le réseau ATM	194

B.1.4. Le modèle de référence du protocole ATM	195
B.1.5. La couche physique	196
B.1.6. La couche ATM	200
B.1.7. La couche d'adaptation à ATM (AAL)	202
B.1.7.1. AAL 1	204
B.1.7.2. AAL 2	205
B.1.7.3. AAL 3/4	205
B.1.7.4. AAL 5	209
B.1.8. Le service en mode sans connexion	211
B.1.9. Les principes d'exploitation et de maintenance	212
B.1.10. La signalisation ATM	214
B.1.11. La gestion du trafic et des encombrements dans les réseaux ATM	215
B.1.11.1. La gestion des ressources (NRM)	216
B.1.11.2. La fonction d'admission et de contrôle (CAC).....	217
B.1.11.3. Le contrôle des paramètres d'utilisation ou de réseau (UPC/NPC) ...	218
B.1.11.3.1. Les mécanismes de surveillance.....	218
B.1.11.3.2. La conformation du trafic.....	219
B.1.11.4. Les commandes de rétroaction	220
B.1.11.5. La gestion des priorités (PC)	220
B.2. L'état de la normalisation ATM	221
B.3. Les travaux de recherche sur ATM	223
B.4. Les recommandations CCITT pour le RNIS à Large Bande	226
B.5. Les spécifications de l'ATM Forum	227
B.6. Les documents de l'IETF	227
B.7. Références relatives à l'annexe B	228
 Annexe C : La carte ATM développée	 231

Liste des Figures

Figure II.1	: La station multimédia du DAN.....	44
Figure II.2	: La machine Micronet	45
Figure II.3	: Les deux configurations possibles (Workstation et Netstation)	46
Figure II.4	: Le système VuNet.....	47
Figure II.5	: La station multimédia de l'Université de Washington	48
Figure II.6	: Le système de l'Université de Lancaster	49
Figure II.7	: La station multimédia de NEC	50
Figure II.8	: Le système de l'Université de Melbourne	51
Figure III.1	: L'architecture générale du communicateur	57
Figure III.2	: Topologie en double anneau.....	58
Figure III.3	: Réduction de la dimension du réseau	58
Figure III.4	: Topologie privilégiant certaines communications	59
Figure III.5	: Structure en bus pour la télé-surveillance d'autoroutes ou de tunnels ...	59
Figure III.6	: Configuration équivalente à la figure III.5 avec des commutateurs externes	59
Figure III.7	: Configuration de la machine pour obtenir un commutateur 4x4	60
Figure III.8	: Configuration équivalente à la figure III.5 avec redondance	60
Figure III.9	: Schéma de la carte fille d'interface physique.....	62
Figure III.10	: Les configurations possibles pour les accès au tampon	66
Figure III.11	: Schéma de la carte mère d'adaptation à ATM.....	68
Figure III.12	: Les copies mémoire lors de la réception sans le micro-contrôleur local	72
Figure III.13	: Schéma de la carte fille micro-contrôleur local	74
Figure III.14	: Les copies mémoire lors de la réception avec le micro-contrôleur local	76
Figure III.15	: Le système Producteurs-Consommateurs classique	79
Figure III.16	: Le système Producteurs-Consommateurs envisagé.....	79
Figure III.17	: Le système Producteurs-Consommateurs virtuel	80
Figure III.18	: L'organisation générale des protocoles et des logiciels	84
Figure III.19	: La structure des protocoles	87
Figure III.20	: La structure détaillée de la couche AAL de signalisation	90
Figure III.21	: Les entrées/sorties d'un commutateur	104
Figure III.22	: Les modèles UNI et NNI	111
Figure III.23	: Le modèle UNI-NNI	112

Figure III.24 : Les échanges de la signalisation UNI	114
Figure III.25 : Les échanges de la signalisation NNI	116
Figure III.26 : L'ouverture de connexions par une tierce partie	118
Figure IV.1 : Le modèle du système de communication	135
Figure IV.2 : Les implantations à comparer	138
Figure IV.3 : Le débit maximum pour une connexion en émission	141
Figure IV.4a : Le délai de bout en bout pour une connexion en émission	142
Figure IV.4b : Zoom sur la figure IV.4a	143
Figure IV.5 : Le rapport du débit maximum pour une connexion en émission.....	143
Figure IV.6 : Le rapport du délai de bout en bout pour une connexion en émission	144
Figure IV.7 : Le débit maximum pour une connexion en émission et une en réception	145
Figure IV.8 : Le délai de bout en bout pour une connexion en émission et une en réception	146
Figure IV.9 : L'évolution des délais en fonction du nombre de connexions	147
Figure V.1 : La comparaison des coûts sans le micro-contrôleur local	159
Figure V.2 : La comparaison des coûts avec le micro-contrôleur local	159

Liste des Tables

Table II.1 : Les paramètres relatifs aux différentes catégories de flux	33
Table II.2 : La comparaison des solutions suivant les caractéristiques recherchées	52
Table III.1 : La capacité du tampon avec 2 descripteurs par connexion (en nombre de cellules	70
Table III.2: La capacité du tampon d'émission et de réception (longueur du message par connexion)	71
Table IV.1 : L'évolution de la période de rafraîchissement	148
Table IV.2 : L'évolution du nombre de variables par connexion	149
Table IV.3 : L'évolution du nombre de variables avec la charge du processeur	150
Table IV.4 : L'évolution du nombre de variables pour un processeur hôte de 100 MIPS	151
Table IV.5 : L'évolution du nombre de variables pour un processeur hôte de 200 MIPS	152
Table V.1 : Les coûts des éléments considérés	158

Chapitre I

Introduction

Aujourd'hui, les réseaux de communication sont en constante évolution, offrant des débits de quelques dizaines de mégabits à quelques gigabits par seconde sur le terrain, et quelques centaines de gigabits en laboratoire. Citons pour ce qui existe dans le commerce FDDI [Ros86], ATM [Puj94], Ethernet 100 Mbit/s, Fiber Channel [Prt93]. D'un autre côté, les applications multimédia demandent des performances de traitement de plus en plus élevées. Ces avancées technologiques sont utilisées pour le secteur des télécommunications et de l'informatique bureautique. Nous nous proposons d'étudier l'impact de la technologie la plus étudiée actuellement, ATM, dans le domaine des applications distribuées temps réel.

L'**objectif général** de cette thèse est de proposer l'architecture d'un système de communication basé sur ATM permettant d'intégrer les flux audio et vidéo dans les applications distribuées temps réel.

Ce travail entre dans le cadre du projet RACINES de l'IMAG, dans lequel sont menées des études d'impact des nouvelles technologies de communication, et en particulier ATM, sur les architectures des systèmes informatiques, qu'ils soient distribués ou parallèles. Le projet de recherche s'est déroulé en collaboration étroite avec l'entreprise ITMI-APTOR de Meylan, filiale du groupe Cap Gemini Sogeti. Ce contact a permis de bien connaître les besoins et les attentes des utilisateurs, et d'étudier de nouveaux concepts tout en proposant des solutions industriellement viables. Outre les investissements propres à l'entreprise en homme et en matériel, les recherches ont bénéficié du soutien :

- Du Ministère de l'Enseignement Supérieur et de la Recherche (MESR), par le biais d'un contrat de recherche.
- De la Région Rhône-Alpes qui m'a octroyé une bourse de recherche pendant ces trois années.
- Et du PRC/GdR Architectures Nouvelles de Machines.

La suite de cette introduction est organisée comme suit. Après avoir présenté la problématique des applications distribuées temps réel, les limites des solutions actuelles sont analysées. L'adéquation de la technique ATM, choisie dans RACINES, aux besoins du monde industriel est ensuite discutée. Finalement, les objectifs et l'organisation de la thèse sont détaillés.

I.1. Les applications distribuées temps réel

Les applications distribuées temps réel sont présentes dans les domaines de la productique et de la gestion technique de bâtiments, dans les centres de commandement, dans les grands sites et les grands automatismes embarqués (bateaux, trains, avions, ...). Ces applications sont caractérisées par des exigences de plusieurs sortes : temps réel, hétérogénéité, disponibilité, sûreté.

Aujourd'hui, les systèmes de communication proposés pour ce type de besoins sont basés sur des aménagements des normes existantes. L'action la plus spectaculaire de ces dix dernières années a été menée par General Motors avec la définition de MAP (Manufacturing Automation Protocol) [Mur87], plus spécialement destiné aux applications de productique du domaine manufacturier. A cette occasion, la norme 802.4 (bus à jeton) a été définie [Rol88]. Cependant, ce sont plutôt des solutions sur des bases Ethernet qui s'imposent sur le terrain. De même, les besoins temps réel exprimés pour les applications de défense (marine) ont fait l'objet de divers travaux, dont GAMT103 en France ou XTP [XTP92] soutenu par la marine américaine. Une approche originale pour les applications distribuées tolérant les fautes a été développée par Bull dans le cadre du projet Delta 4. En France, un Ethernet déterministe a été défini par l'INRIA afin de répondre aux contraintes temps réel.

Dans la couche application, la norme MMS (Manufacturing Message Specification) [Bri91] est un résultat de l'opération MAP. Bien que pour les automaticiens, MMS comporte probablement trop de concepts issus de l'informatique (objets, tâches, événements) et pas assez d'éléments de contrôle effectif du temps (délais, cycles, temps de réponse, débits, synchronisations fines), cette technologie est bien acceptée pour la construction des applications distribuées temps réel, puisque c'est la seule normalisée sur le marché des applications visées.

Mais, les besoins de ces applications sont dès aujourd'hui poussés par les fonctions de télé-surveillance (images), de communications interpersonnelles (sons), de synchronisations globales et d'horlogerie répartie.

De plus, l'utilisation de plusieurs réseaux dédiés à un seul type de données est inévitable actuellement. On trouve dans la plupart des cas un réseau pour les applications de bureautique, un pour le téléphone, un pour les alarmes incendies, un pour l'échantillonnage et les commandes des capteurs/actionneurs, voire un autre pour la gestion technique de bâtiments.

Par conséquent, il devient de plus en plus difficile à servir les besoins avec les technologies utilisées aujourd'hui. Les limites des solutions actuelles sont énumérées dans le paragraphe suivant.

I.2. Les limites des solutions actuelles

L'accroissement du nombre d'applications qui mettent en oeuvre à la fois les équipements informatiques/automatiques classiques et des équipements audio/vidéo rend les infrastructures de communication de plus en plus complexes et de moins en moins évolutives. La multiplicité des câblages pour supporter les différents flux de données provoque des difficultés de mise en place et de maintenance.

De plus, les réseaux locaux informatiques actuels tels Ethernet ou FDDI supportent mal les flux audio/vidéo. En effet, ce type de données nécessite à la fois une bande passante élevée et des contraintes de synchronisation importante. Peu de réseaux supportent à la fois des contraintes isochrones et des paramètres de qualité de service. Certaines études ont été menées pour rendre certains réseaux existants isochrones. Mais elles restent limitées, bien qu'un Ethernet isochrone ait été défini.

D'autre part, du point de vue des protocoles de communication, l'empilement des couches de protocoles et leur exécution systématique en logiciel ne sont pas toujours optimums pour les traitements temps réel. Que ce soit le modèle OSI ou des protocoles tels que TCP/IP, la multitude d'étapes à exécuter entre la ligne physique et l'application provoque des délais importants, parfois difficilement maîtrisables.

Le nombre de copies mémoire est un des points cruciaux dans le délai subi par les données échangées par le réseau. Très souvent, ce nombre est élevé dans les systèmes actuels. Par exemple, lors de la réception d'un message, celui-ci est d'abord stocké dans le tampon de réception de la carte d'interface au réseau. Suivant la solution, il peut être recopié dans une autre mémoire sur la carte d'interface, avant d'être transféré dans la mémoire de la machine hôte. En fonction du protocole de communication concerné, le système opère une ou plusieurs nouvelles copies avant de le délivrer à l'application. L'ensemble de ces manipulations peut peser très lourd dans le temps d'acheminement d'un message.

Toutes ces limites nous ont conduits à étudier l'adéquation de la technique ATM aux besoins des nouvelles applications distribuées temps réel.

I.3. L'adéquation de la technique ATM aux applications distribuées temps réel

La technique ATM (Asynchronous Transfer Mode) [CN91] a vu le jour en France, au CNET, au début des années 80. Elle a été retenue quelques années plus tard par le CCITT (Comité Consultatif International Télégraphique et Téléphonique, aujourd'hui appelé Union Internationale des Télécommunications (UIT-T ou ITU-T)) comme base du Réseau Numérique à Intégration de Services Large Bande (RNIS-LB ou B-ISDN). Elle était au début destinée au monde des réseaux publics grande distance. Devant la perspective d'un développement au niveau mondial par les opérateurs de télécommunication, elle est reprise aujourd'hui par les constructeurs informatiques pour l'adapter aux réseaux privés. L'ATM Forum, regroupant plus de 500 industriels, fournit de gros efforts pour faire avancer rapidement la normalisation.

ATM [Pr91, Bo92] se caractérise par :

- L'utilisation de liens séries rapides (155 et 622 Mbit/s voire 2,5 Gbit/s dans le futur).
- Une technique de transport de l'information basée sur des cellules de taille fixe de 53 octets.
- L'utilisation de commutateurs avec un maillage physique quelconque.
- Des communications basées sur des connexions virtuelles.
- Le traitement de tous types de données, aussi bien isochrones (son et image) qu'asynchrones (données).

Outre l'aspect normalisation, qui permettra à moyen terme de disposer de composants à prix réduits, ATM possède des caractéristiques réseau intéressantes pour le domaine qui nous concerne.

En effet, les applications distribuées temps réel utilisent à 90 % des schémas de communication en mode connecté, établis à la configuration. De plus, les connexions virtuelles ou leur succession dans le temps est connue statiquement à l'avance, pour une application donnée.

Contrairement aux applications de type télécommunication, les trafics ne sont pas aléatoires, les applications distribuées ayant plutôt un comportement global synchrone : acquisition de données, calcul et communication des résultats. L'infrastructure de connexions calibrées offerte par ATM convient, à condition d'opter pour un contrôle strict de débit. Les risques de congestion sont par contre à étudier.

Les informations sensibles de contrôle/commande (automates, robots, ...) doivent être favorisées dans leur temps d'acheminement, par rapport aux autres. Les applications

cibles sont en effet sujettes aux phénomènes d'avalanches d'alarmes, et sensibles aux problèmes de péremption des informations. Le partage de bande passante d'ATM offre des solutions plus simples à mettre en oeuvre que la gestion de priorité dans les réseaux locaux classiques. De plus, ATM étant basé sur un médium non partagé, contrairement à Ethernet, il sera plus aisé de garantir le temps d'accès au réseau.

Les informations de nature isochrone (son, image) sont théoriquement transportées plus facilement puisque ATM est à la base multimédia.

Il reste évidemment à répondre aux communications en mode non connecté, pour les 10% résiduels de communications. ATM n'est pas bien adaptée à ce besoin et il faut en particulier supporter les protocoles de communication utilisés aujourd'hui, tels TCP/IP.

Du point de vue de la disponibilité, il est nécessaire de gérer la redondance de chemins de communication au niveau d'un coupleur, les autres niveaux pouvant être traités de façon traditionnelle. Les mécanismes de connexions virtuelles et de reconfiguration des chemins permettent de supporter ce service.

D'autre part, l'utilisation d'ATM ouvre la voie à la connexion d'équipements nouveaux, pour le son et l'image. On peut imaginer utiliser des caméras comme capteurs d'images, fonctionnant de manière similaire aux capteurs de température ou de données d'autres types. Le traitement de l'image peut alors être déporté sur des machines distantes.

De plus, la perspective du déploiement d'une infrastructure de réseau public basée sur la technologie ATM permet d'envisager des applications largement distribuées au travers de réseaux grandes distances. Aujourd'hui, l'hétérogénéité des solutions rend difficile de telles applications du fait des retards conséquents introduits par les passerelles à traverser. L'emploi de la même technique dans les réseaux locaux et dans les réseaux publics devrait réduire les délais d'acheminement.

Toutes ces constatations nous ont conduits à fixer les objectifs définis dans le paragraphe suivant afin d'obtenir un système de communication performant.

I.4. Les objectifs

L'objectif de ce travail est de concevoir un système de communication complet, basé sur ATM, permettant de répondre aux besoins rencontrés dans les applications distribuées temps réel. Le système doit ainsi pouvoir supporter des équipements aussi divers que des caméras, des écrans, des calculateurs, des robots, des automates, des microphones, des capteurs ou des actionneurs. Outre les contraintes temps réel typiques de cette catégorie de réseaux, il s'avère nécessaire d'intégrer aujourd'hui les flux audio et vidéo au même

titre que les données informatiques classiques. De plus, le même réseau doit supporter tant les flux temps réel que de bureautique, comme défini dans le concept d'usine intégrée (CIM : Computer Integrated Manufacturing) [Lep89].

Par conséquent, le système de communication envisagé doit avoir les caractéristiques citées ci-après :

- C1 Supporter plusieurs équipements audio/vidéo sur une même station (ce qui se traduit par un besoin de bande passante élevée au coeur de la station), afin de limiter au maximum l'emploi de matériel encombrant.
- C2 Permettre de construire des topologies variées et évolutives, afin de s'adapter aux différentes utilisations possibles (système embarqué, câblage d'usines ou d'ateliers, ...). En général, les systèmes centralisés sont mal adaptés pour des problèmes de robustesse et de difficulté de câblage.
- C3 Offrir la redondance des chemins d'accès afin d'assurer une disponibilité maximale.
- C4 Garantir des qualités de service aux applications temps réel, par exemple pour assurer que les alarmes arriveront à temps ou assurer des cycles de contrôle-commande courts.
- C5 Supporter les applications de téléinformatique classique pour récupérer l'existant et offrir une administration standardisée, afin d'utiliser les outils du marché.
- C6 Avoir un coût peu élevé.

I.5. Le plan de la thèse

Pour définir le système de communication, il nous faut étudier plus particulièrement deux points :

- Une architecture de couplage d'équipements performante.
- Une gestion des communications efficace entre les équipements.

Le chapitre deux dresse un état de l'art sur l'architecture des machines pour les hauts débits et permet de mieux cerner les problèmes posés par l'emploi des nouvelles technologies de communication. Les nombreuses études dans ce domaine fournissent des solutions où la machine est entièrement repensée et d'autres avec seulement des cartes d'interface au réseau.

Le chapitre trois présente l'architecture du système de communication, avec d'une part les choix de conception matérielle et logicielle pour le couplage des équipements, et d'autre part les caractéristiques particulières des communications entre les équipements.

Dans le chapitre quatre, une étude de performances est réalisée. Elle apporte une première idée des caractéristiques que l'on peut attendre de cette architecture.

Le chapitre cinq évalue le rapport performances/coût de l'architecture de communication proposée et le compare aux solutions existantes.

La conclusion rappelle les originalités des solutions choisies et cite les points qui devront faire l'objet d'études plus approfondies dans l'avenir.

Chapitre II

Les grands projets dans l'architecture des machines pour les hauts débits

Les débits offerts par les réseaux augmentent rapidement et atteignent aujourd'hui quelques gigabits par seconde, voire plusieurs dizaines en laboratoire. Ils ouvrent de nouvelles perspectives pour les applications distribuées. Néanmoins, les réseaux haut débit soulèvent de nouveaux problèmes de conception, tant pour les machines que pour les interfaces au réseau. Ce chapitre s'intéresse dans une première partie à la conception d'architectures capables de supporter les débits élevés, en insistant sur les points à étudier avec soin pour réaliser ces systèmes. La conception des interfaces de communication entre le réseau et la machine est ensuite présentée, illustrée par de nombreuses réalisations. La troisième partie décrit des projets dans lesquels l'architecture de la machine a été entièrement repensée pour traiter les flux haut débit.

II.1. La conception d'architectures pour les hauts débits

L'interface entre le réseau et la machine, et l'implantation des protocoles de communication sont des sujets de recherche importants qui n'ont pas été pris en considération lors de l'apparition des réseaux de communication. A cette époque, les études portaient plus sur le partage des liens de communication. Il y a deux raisons à ce manque d'intérêt pour ces sujets : les vitesses des liens étaient faibles et les applications disponibles comme la connexion à distance ('telnet') ou le courrier électronique n'avaient pas de contraintes particulières en terme de débit ou de délai.

Aujourd'hui, les réseaux haut débit permettent de supporter de nouvelles applications distribuées, à grande distance ou multimédia. Ces dernières imposent de nouvelles contraintes sur l'architecture de communication, tant au niveau de l'interface avec le réseau, que des machines hôtes et de leur système d'exploitation. En particulier, elles nécessitent des débits élevés jusqu'au cœur de la station et des délais garantis. Mais, la conception actuelle des stations de travail et l'implantation des protocoles de communication limitent fortement la bande passante disponible pour les applications. En effet, dans la plupart des systèmes, le processeur de la station est au centre de l'organisation de la transmission de l'information temps réel sur le réseau. Ce genre d'architecture est mal adapté au multimédia du fait de l'utilisation intensive des entrées/sorties.

Pour concevoir un système de communication apte à supporter des débits élevés, tous les acteurs doivent être considérés. Ceci inclut l'interface réseau, les couches de protocole, les transferts mémoire, l'utilisation des bus, le système d'exploitation et l'architecture de la machine hôte. De nombreuses techniques sont apparues afin d'optimiser le système : limiter ou éliminer les copies mémoire et les interruptions vers le

processeur hôte, minimiser le traitement des protocoles sur les données ("lightweight protocols"), exécuter la plus grande partie du protocole sur la carte d'interface au réseau. Dans la suite de ce paragraphe, nous présentons les besoins des nouvelles applications envisageables avec l'apparition des débits élevés, puis les solutions en termes de protocoles de communication.

II.1.1. Les contraintes des nouvelles applications

Pendant longtemps, les réseaux informatiques ont transporté des données sans contraintes temporelles particulières. Le but était d'assurer un transport fiable des informations, par exemple pour le transfert de fichiers ou la connexion à distance. Pour répondre aux besoins de certaines applications nécessitant des contraintes temps réel, des adaptations spécifiques à chaque problème ont été apportées aux réseaux.

Aujourd'hui, avec les réseaux haut débit, de nouvelles applications apparaissent, intégrant en particulier les flux de son et d'images animées. Ce sont principalement les applications multimédia comme la télé-conférence, le télé-enseignement, la télé-surveillance, le travail coopératif [Bry92], l'imagerie médicale, la diffusion de films ou de programmes de télévision, la réalité virtuelle [Par93]. D'autres applications sont envisageables, par exemple le calcul scientifique réparti.

Un des besoins principaux de ces applications est la garantie de qualité de service [Vog94, Taw92, Bie90]. Les paramètres importants à considérer pour les applications haut débit sont :

- Le délai du réseau : des applications ont besoin de spécifier le temps maximum pour que les données arrivent à l'application distante. Ceci est particulièrement important pour les applications multimédia temps réel où l'utilisateur peut être sensible aux délais élevés (par comparaison, les conversations téléphoniques passant par satellite subissent un délai perceptible).
- Le débit : chaque application doit pouvoir demander le débit dont elle a besoin.
- La fiabilité : des applications peuvent être sensible à la perte de données (par exemple, le transfert de fichiers n'accepte pas la moindre perte).
- Le temps d'établissement du service : pour les communications courtes, il est important de connaître le temps qui s'écoule avant le transfert effectif des informations.
- Le temps d'interarrivée ou gigue : les applications telles que le son ou la vidéo sont très sensibles au temps écoulé entre les arrivées successives de données. Sans un débit de données régulier, le son risque d'être haché, voire incompréhensible.

La table I.1 donne les valeurs des paramètres définis ci-dessus pour les différents flux de données susceptibles d'être transportés. Ces valeurs sont tirées de [Heh90] et [Pry91]. Pour la vidéo et le son, elles sont basées sur la capacité de l'individu à ne pas voir (ou à ne pas entendre) les erreurs.

Service	Débit moyen (Mbit/s)	Délai maximum (s)	Gigue maximum (ms)	Taux d'erreur par bit	Taux d'erreur par message
Voix non compressée	0,064	0,25	10	$< 10^{-1}$	$< 10^{-1}$
Son haute qualité non compressé	0,32 à 2	0,25	10	10^{-6}	10^{-8}
Vidéo standard non compressée	20 à 30	0,25	10	10^{-2}	10^{-3}
TV haute définition non compressée	100 à 150	0,25	1	10^{-6}	10^{-8}
Vidéo compressée	1 à 30	0,25	1	10^{-6}	10^{-9}
Images	1 à 10	1	-	0 à 10^{-4}	0 à 10^{-9}
Transfert de fichiers	1 à 100	1	-	0	0
Données temps réel	< 10	0,001 à 1	-	0	0

Table II.1 : Les paramètres relatifs aux différentes catégories de flux

En résumé, les nouvelles applications demandent des débits élevés jusqu'au cœur de la machine hôte, et ont besoin d'une garantie de qualité de service pour leurs transferts. Le paragraphe suivant décrit différentes études sur l'implantation de protocoles de communication pour supporter les nouvelles applications.

II.1.2. L'implantation des protocoles de communication

Aujourd'hui, les couches OSI 1 et 2-MAC des protocoles de communication sont câblées. D'autre part, il est universellement accepté que les couches session et présentation sont coûteuses [Cla90]. Par conséquent, elles sont très rarement utilisées. Cependant, une alternative à la syntaxe ASN.1 [Kal91] a été définie dans [Dab92]. Le problème se situe donc au niveau des couches réseau et transport [Dos92]. Ces couches sont généralement implantées en logiciel, d'où des temps de traitement élevés. Afin de délivrer les flux haut débit aux applications et supporter les contraintes de qualités de service, différentes solutions ont été proposées. Elles concernent :

- L'optimisation du logiciel.
- L'implantation câblée des protocoles de transport.
- La définition de nouveaux protocoles.

Le plus gros progrès dans l'amélioration des protocoles en logiciel a été réalisé par Van Jacobson [Jac90, Cla89] sur TCP/IP. Il a opéré trois changements majeurs dans l'implantation sous UNIX BSD 4.2 afin d'améliorer les performances. Le premier et le plus important a été de supprimer les copies redondantes de données dans le système d'exploitation. Le second a consisté à augmenter la taille des paquets échangés à l'interface entre le noyau et le pilote. Le troisième a modifié l'algorithme de traitement de l'en-tête TCP pour prendre en compte la destination du dernier paquet reçu. Cette modification part de la constatation que deux paquets consécutifs ont souvent la même destination. Ces changements ont permis d'augmenter le débit de 2 à 8,4 Mbit/s sur un réseau Ethernet à 10 Mbit/s, avec des paquets de 32 Ko.

D'autres moyens pour essayer d'améliorer les performances ont été proposés. L'un d'eux consiste à réduire la complexité du protocole de transport. URP (Universal Receiver Protocol) est un des premiers projets dans le domaine. Il part du constat que la réception est le goulot d'étranglement des protocoles, et s'attache à simplifier le traitement de la réception. La taille du paquet est petite pour réduire le délai dû au mécanisme de "store-and-forward".

D'autres protocoles de transport réduits ont été proposés. C'est le cas du projet XTP (Xpress Transfer Protocol) [XTP92]. Il a été conçu pour être implanté sur une architecture VLSI, appelée Protocol Engine. Il a pour objectif de fournir toutes les fonctions d'un protocole de transport, de supporter des contraintes temps réel, et de fonctionner à haut débit (1 Gbit/s). A cette fin, il utilise un schéma de retransmission sélectif et un reséquencement optionnel des données.

NETBLT (NETwork BLock Transfer) [Cla87] transmet des groupes de paquets afin d'améliorer les performances. Cependant, il supporte mal les paquets de petite taille.

VMTP (Versatile Message Transaction Protocol) [Che89] est un protocole de transport qui offre des services de communication performants pour les systèmes distribués. Il supporte les transactions temps réel.

Des solutions intermédiaires ont vu le jour. Ainsi, la parallélisation des traitements a été étudiée. Le système de transport Patrocolos [Brz92], inspiré de XTP, a été conçu pour fournir une implantation efficace sur des architectures multiprocesseurs, tels que des réseaux de Transputer combinés avec du matériel spécialisé pour les fonctions critiques en temps de traitement.

Une approche originale a été proposée dans [Haa91]. Partant de la constatation que les couches de protocoles effectuent des opérations redondantes, Haas propose une architecture où le protocole de communication peut s'adapter à tous les protocoles de niveau 4 à 6. Le protocole est composé de trois niveaux : Network Access Control (couches 1 à 3), Communication Interface (couches 4 à 6) et Application. La couche CI est constituée d'un ensemble de fonctions (contrôle de flux, retransmission, présentation, adressage, ...) s'exécutant en parallèle. L'utilisateur choisit les fonctions pour construire son protocole. Afin de décharger le processeur hôte au maximum, la couche CI est câblée.

Pour supporter les contraintes des nouvelles applications, il ne suffit pas de traiter les informations à débit élevé, il faut aussi offrir des garanties de qualité de service. Plusieurs protocoles de transport ont été définis à cette fin. Les implantations d'un protocole de transport au-dessus de ST-II [Mit94] réalisé par BERKOM [Boe94] ou l'Université d'Heidelberg [Del92] apportent des réponses en terme de paramètres de qualité de service (débit, délai de transit de bout en bout, classes de confiance) ou de synchronisation entre l'émetteur et le récepteur. TP++ développé par Bellcore [Fel93] est un protocole de transport haut débit (1 Gbit/s) pour les applications multimédia, dans lequel le multiplexage de plusieurs connexions est interdit pour supporter plus efficacement la qualité de service. L'Université de Lancaster a une approche différente qui consiste à supporter la qualité de service au niveau du système d'exploitation (micro-noyau Chorus) [Rob94, Bla94].

II.1.3. Résumé

En résumé, de nombreuses solutions ont été proposées pour implanter efficacement les protocoles de transport afin de supporter des débits élevés et offrir une garantie de qualité de service. Mais, sans une interface au réseau efficace, ces efforts sont anéantis. Le paragraphe suivant s'intéresse au problème de la conception des interfaces entre le réseau et la machine pour les hauts débits.

II.2. Les interfaces réseau

L'interface entre le réseau et la machine hôte est le premier goulot d'étranglement dans la communication. Elle comprend deux parties : l'adaptateur réseau (partie matérielle) et le logiciel. Deux problèmes principaux se posent dans la conception d'une interface réseau. Le premier concerne le partage des fonctions du protocole entre l'interface réseau et le processeur hôte. Le deuxième est relatif à la complexité de la carte

d'interface (présence d'un processeur, quantité de mémoire, type de communication avec le processeur hôte).

En général, l'interface réseau prend place dans un connecteur, sur le bus fond de panier de la machine, plutôt que directement sur le bus local de la mémoire [Ram93]. Une solution efficace avec une interface s'enfichant dans un connecteur de mémoire SIMM a toutefois été proposée dans [Min94].

Le paragraphe est organisé comme suit : le problème des copies mémoire et des transferts sur le bus est d'abord présenté. C'est un des problèmes principaux pour le support de débits élevés. Puis la répartition des protocoles entre la carte d'interface et la machine hôte est analysée, en soulignant dans chaque cas les avantages et les inconvénients. Finalement, l'intégration de l'interface avec le système d'exploitation de la machine est décrite. Ce dernier point est aussi à considérer puisque le logiciel est source d'un ralentissement non négligeable des traitements.

II.2.1. La réduction des copies et des transferts sur le bus

Aujourd'hui, le débit d'application à application que les stations peuvent supporter est de quelques mégabits par seconde, même si le débit du réseau est de 100 Mbit/s ou plus. Comme les réseaux deviennent plus rapides, les opérations affectant tous les octets d'un message comme les copies ou les calculs de codes détecteurs d'erreurs dominent pour deux raisons. Premièrement, les messages transportés sont plus grands, et par conséquent la partie fixe du temps de traitement par paquet est plus facilement amortie. Deuxièmement, la mémoire du système est souvent la partie la plus critique de la machine, et les opérations sur les octets font largement intervenir cette ressource. En effet, si la puissance des processeurs augmente rapidement et régulièrement, les temps d'accès aux mémoires restent quasiment stables.

Dans les systèmes traditionnels supportant TCP/IP, le nombre de transferts sur le bus mémoire de la machine hôte peut s'élever à 6 [Ste94]. Dans cette situation, en émission, l'application écrit le message dans un tampon de son espace d'adressage. Ensuite, le programme associé au "socket" copie les données dans un tampon du système, et le protocole de transport lit les données pour calculer le code de détection d'erreurs. Finalement, la couche liaison copie les données sur l'adaptateur réseau. Dans certaines implémentations, une copie supplémentaire combine les données de plusieurs tampons système dans un seul tampon relatif à la carte, ce qui ajoute deux transferts sur le bus.

La copie des données entre l'interface réseau et la machine hôte ne peut être évitée car les applications ont besoin de lire leurs données dans leur mémoire locale. Cependant, plusieurs autres déplacements de données sont effectués à l'intérieur du système d'exploitation [Tan93]. Chaque fois que les données traversent la frontière d'un domaine (par exemple le noyau ou des couches de protocoles), une copie est effectuée, de manière à transformer la structure de données dans un format adapté au domaine suivant. Sachant que plusieurs frontières peuvent être traversées à l'intérieur même du noyau du système d'exploitation, le mouvement de données devient un point crucial à traiter afin de concevoir un système de communication performant.

Pour réduire le nombre de copies et de transferts sur le bus, les optimisations se situent à deux niveaux, logiciel et matériel, même si les deux sont étroitement liés puisque des améliorations apportées au matériel impliquent des modifications du logiciel. Dans un premier temps, les améliorations logicielles seront examinées. Ensuite, les différentes optimisations matérielles seront passées en revue.

II.2.1.1. Les optimisations logicielles

Le type d'API (Application Program Interface) réseau utilisé détermine la difficulté pour effectuer des optimisations. Dans le cas de l'interface socket [Tan90], en émission, l'application spécifie la zone dans laquelle les messages sont construits, et les opérations sont bloquantes jusqu'à ce que la zone soit libérée. Ceci implique que le système opère une copie des données lors de l'appel de la primitive "send" au cas où une retransmission serait nécessaire, sinon l'application serait bloquée jusqu'à ce que les données soient acquittées. En réception, l'application spécifie où les données doivent être stockées. Ainsi, les données doivent être copiées durant l'opération "receive". Ces primitives sont qualifiées d'immédiates par [Ste94], par opposition aux primitives tamponnées.

Avec les primitives immédiates, le calcul du code de détection d'erreurs peut être effectué pendant la copie, éliminant ainsi une copie. Par contre, pour éliminer d'autres transferts, il faut modifier l'API. En effet, une interface de type "socket" nécessite deux transferts au minimum. Les systèmes tels TCP ou XTP utilisent les primitives immédiates.

Avec les primitives tamponnées, l'application construit et reçoit les messages dans des tampons partagés avec le système. La gestion des tampons est réalisée par des pointeurs échangés entre l'application et les primitives réseau. Ceci permet d'éliminer une à deux copies en émission comme en réception. La propriété du tampon est transférée au système lors de l'émission, ce qui évite de faire une copie pour assurer la retransmission. Par contre, l'application ne peut plus utiliser ces données dès qu'elle les a envoyées. Les primitives tamponnées sont utilisées dans différents systèmes tels que Nectar [Arn89] ou le VMP Network Adapter Board (NAB) [Kan88].

Les primitives tamponnées ont deux inconvénients par rapport aux primitives immédiates : pour des protocoles comme TCP qui ont le code détecteur d'erreurs dans l'en-tête, il est nécessaire d'effectuer une lecture de toutes les données pour le calculer, séparée de l'opération de copie. Les protocoles comme XTP qui stocke le code de détection d'erreurs en queue de message résolvent ce problème. Le deuxième inconvénient est que l'application est plus sollicitée puisqu'elle doit gérer des zones mémoires qui contiennent les messages reçus ou à envoyer, et attendre leur libération. Mais c'est cette flexibilité accrue donnée au système (et par voie de conséquence cette diminution de flexibilité pour l'application) qui permet aux primitives tamponnées d'être plus efficaces.

II.2.1.2. Les optimisations matérielles

Les solutions matérielles pour optimiser les transferts sur le bus sont au nombre de trois :

- Ajouter un DMA (Direct Memory Access) sur l'interface réseau pour améliorer les mouvements de données.
- Placer un tampon partagé avec le système sur l'adaptateur réseau pour éliminer des copies.
- Exécuter le protocole sur la carte réseau.

Les deux premières solutions sont analysées dans la suite de ce paragraphe. La troisième fait l'objet du paragraphe suivant qui discute la répartition de l'exécution des protocoles entre interface réseau et machine hôte.

Deux techniques permettent de transférer les données depuis la carte d'interface au réseau vers la mémoire de la machine hôte :

- Le processeur hôte déplace les données (cette méthode est appelée mode programmé ("Programmed I/O")).
- Un DMA est chargé de cette tâche.

Le DMA permet de déplacer les données entre la carte réseau et la machine hôte sans intervention du processeur hôte, sauf pour l'initialisation du transfert qui demande de fournir l'adresse et la longueur des données. Il offre deux avantages par rapport au mode programmé :

- Le DMA nécessite généralement un seul transfert sur le bus, alors que le mode programmé en demande deux.
- Le processeur peut travailler en parallèle avec le DMA. Ceci dépend cependant énormément de la localité des données dans le cache du processeur. En effet, si le processeur a besoin de données dans la mémoire centrale, il doit partager le bus avec le transfert DMA.

Par contre, l'utilisation du DMA introduit de nouveaux problèmes, et ses performances dépendent des conditions d'utilisation.

Le DMA introduit en premier lieu des problèmes de cohérence de cache. Lors de l'émission, il est nécessaire d'opérer la mise à jour de la mémoire pour les blocs à transférer avant d'effectuer l'envoi. En réception, il peut être nécessaire d'invalider les valeurs du cache lorsque celles-ci sont mises à jour par les données en provenance du réseau. Cette invalidation, généralement effectuée par le système d'exploitation, est coûteuse en temps. Aujourd'hui, des invalidations matérielles apparaissent sur les processeurs, ce qui permettra de rendre le transfert DMA plus efficace.

Contrairement au mode programmé, le transfert DMA est asynchrone et le processeur doit être averti de la fin d'un transfert, généralement au moyen d'une interruption. Les interruptions sont une source importante de diminution des performances au niveau du processeur les traitant.

La taille des blocs de données transférées joue aussi un rôle important. En effet, pour des transferts courts, le DMA n'est pas très efficace puisque le temps d'initialisation est important. Par contre, il devient très rentable pour transférer des blocs de grande taille. De plus, les bus haut débit comme PCI [PCI93] ou Turbo Channel permettent de transférer les données en rafale (une adresse puis un ensemble de données) afin d'offrir des taux de transferts élevés. Le DMA est bien adapté et permet alors d'obtenir des rendements importants.

Des comparaisons de ces différentes approches, et des analyses de l'efficacité du transfert DMA sont fournies dans [Ram93]. En moyenne, ce mode de transfert est efficace. D'ailleurs, la plupart des interfaces proposées pour supporter des hauts débits utilisent ce mode. C'est le cas par exemple du NAB de Stanford [Kan91], de l'interface ATM de Bellcore [Dav91] ou encore de celle de l'Université de Pennsylvanie [Tra92].

La deuxième solution matérielle pour améliorer les performances est l'ajout d'un tampon sur l'adaptateur réseau, en remplacement des FIFO généralement utilisées [Bia93]. Dans ce cas, l'espace mémoire doit être suffisamment grand pour être utilisé comme tampon de retransmission et de réception pour le système. Déporter les tampons du système sur la carte réseau permet d'éliminer la copie due au déplacement des données

entre l'espace utilisateur et l'espace système. Pour les protocoles comme TCP, à l'émission, le code de détection d'erreurs peut-être calculé pendant la copie. En réception, la vérification est facile puisque les données sont disponibles immédiatement dans le tampon, sans attendre la copie dans l'espace utilisateur. Ceci nécessite de calculer le code détecteur d'erreurs en matériel. En combinant cette méthode avec le DMA, le nombre de transferts sur le bus est réduit à deux. Par rapport à la première solution qui nécessitait six transferts (voire huit), le gain est significatif. De plus, les interruptions sont moins nombreuses.

Dans tous les cas, il est nécessaire d'effectuer un transfert sur le bus pour délivrer les données à l'application. Le système Nectar [Arn89] a atteint cette limite. Pour ce faire, il utilise un tampon sur la carte réseau et une interface tamponnée, et l'application est autorisée à accéder directement au tampon externe.

Ce paragraphe a montré les méthodes matérielles pour réduire les transferts sur le bus de la machine hôte. Cependant, suivant la méthode utilisée, le protocole doit s'exécuter sur la machine hôte ou sur la carte d'interface. Le paragraphe suivant examine cet aspect.

II.2.2. La répartition de l'exécution des protocoles entre la carte d'interface et la machine

La répartition de l'exécution des protocoles entre le processeur hôte et la carte d'interface au réseau doit être réalisée de manière réfléchie, car elle peut affecter considérablement les performances [Dav94]. Trois solutions sont répertoriées dans la littérature :

- Le protocole est exécuté sur le processeur hôte. C'est la solution la plus courante.
- Le protocole est exécuté sur le processeur hôte, mais l'interface réseau réalise certaines opérations spécifiques comme le calcul ou la vérification du code de détection d'erreurs, ou la séparation de l'en-tête et de la queue du message.
- Le protocole est exécuté sur la carte d'interface au réseau. Cette exécution peut être prise en charge par un processeur ou par du matériel spécifique [Kan91].

L'utilisation d'un processeur embarqué sur la carte de communication a des avantages non négligeables, tant du point de vue des performances que des services pour la programmation des applications :

- Elle décharge le processeur hôte.

- Elle sépare les tâches réseau des applications. Ainsi, elle permet de prévoir le temps d'exécution, ce qui est indispensable pour les applications temps réel.
- Elle réduit le nombre d'interruptions transmises au processeur hôte.

Cependant, comme tout système, il a aussi des inconvénients :

- La conception de l'interface avec la machine hôte est plus complexe, ainsi que la maintenance du système.
- Le support de protocoles multiples est plus difficile à mettre en oeuvre.
- Il est plus coûteux. Ceci est dû en particulier à la nécessité d'un processeur de performances équivalentes au processeur hôte [Coo90, Ram93].

J.R. Michel a conclu à l'issue de sa thèse sur ce sujet [Mic93] que l'utilisation d'un processeur externe est utile dans deux cas :

- Lorsque le processeur hôte ne peut pas supporter la charge imposée par le traitement du protocole ou les besoins des applications
- Lorsque le protocole doit fournir un niveau de service supérieur à celui que peut offrir la machine hôte et le système d'exploitation. En particulier, dans le cas où le protocole ajoute une charge ou des tâches avec un temps d'exécution non déterministe sur un processeur hôte devant exécuter des applications temps réel, l'emploi d'un processeur externe est indispensable pour supporter le service requis.

II.2.3. L'intégration avec le système d'exploitation

Pour être efficace, le choix du matériel s'accompagne de modifications sur l'approche traditionnelle des couches de protocoles et de l'interaction avec le système d'exploitation. Par exemple, l'utilisation d'un DMA transforme une simple copie en une opération asynchrone. Il est alors nécessaire de resynchroniser le processeur à la fin du transfert, généralement au moyen d'une interruption.

Le gestionnaire de la carte d'interface au réseau est très important du point de vue des performances. Il peut être plus ou moins complexe en fonction du type d'interface choisi : primitives immédiates, primitives tamponnées, tampon partagé sur la carte. Par exemple, l'utilisation de primitives tamponnées est plus efficace car elle donne un plus grand contrôle au système pour le placement et le transfert des données. En conséquence, son implantation est plus difficile.

D'autres solutions ont été proposées pour améliorer les performances, en assurant une bonne interaction entre le système d'exploitation et la carte d'interface. Ainsi, la mise en place de chiens de garde à chaque émission pour éviter d'attendre la réponse de l'interface en cas d'erreur a été proposée. D'après les mesures effectuées dans [Tra94], cette méthode ne semble pas très efficace. D'autre part, la définition de niveaux de priorité pour les interruptions pour réduire la surcharge due à l'accès à la section critique (exclusion mutuelle) a été étudiée par [Sto93].

II.2.4. Résumé

Ce paragraphe a présenté quelques méthodes pour améliorer les performances des interfaces réseau afin de supporter les débits élevés. Il s'avère que l'optimisation du matériel a des incidences importantes sur le logiciel. Néanmoins, la minimisation des transferts sur le bus de la machine hôte est le point le plus important.

En résumé, la solution la plus efficace est celle avec un processeur externe et un tampon sur la carte d'interface. Dans les meilleures conditions, une seule copie est alors possible. Un compromis entre efficacité, complexité et coût est à faire dans chaque situation.

II.3. Les machines de communication

Certains projets ne se sont pas contentés de définir une interface optimale pour la communication avec le réseau, ils ont entièrement repensé l'architecture de la machine. Les architectures proposées étendent les concepts utilisés dans les réseaux haut débit à la conception des machines de communication. Elles visent principalement à intégrer le réseau à l'intérieur de la machine. Le bus fond de panier de la machine est partiellement ou complètement remplacé par un réseau. Ceci est possible aujourd'hui car les caractéristiques des réseaux locaux et des bus d'entrées/sorties convergent et sont très proches [Sac94]. Les paragraphes suivants présentent huit projets d'architecture de machines pour les communications haut débit et multimédia, répartis en trois groupes :

- Ceux qui remplacent le bus fond de panier par un réseau.
- Ceux qui extraient les flux audio/vidéo au plus près du réseau.
- Ceux qui ajoutent un bus pour le traitement audio/vidéo.

II.3.1. Le remplacement du bus fond de panier par un réseau

II.3.1.1. Desk Area Network

Le premier système a été développé à l'Université de Cambridge (Royaume-Uni) au début des années 90 [Hay91, Hay93]. Dans le DAN (Desk Area Network), le bus de la station multimédia est remplacé par un réseau ATM. Un commutateur ATM, provenant du projet Fairisle [Les91], interconnecte les différents éléments de la machine entre eux et ceux-ci avec le réseau ATM (cf. figure II.1). Les éléments sont aussi bien des périphériques (disque, écran, clavier, caméra, microphone, haut parleur) que des composants intrinsèques à la machine (processeur et cache externe, mémoire primaire) [Bar94]. Chaque élément est équipé d'une interface ATM. Selon les concepteurs, cette interface n'est pas plus compliquée qu'un bus standard. Néanmoins, le développement d'une carte spéciale comportant une interface ATM pour chaque composant du système (processeur, mémoire, périphérique de stockage, audio, vidéo, ...) représente un coût significatif. Par exemple, pour la caméra, une carte d'acquisition particulière a été définie. Elle permet de convertir le signal analogique en signal numérique compressé transmis sous forme de cellules vers le commutateur ATM. Cette interface contient des processeurs de traitement du signal, des tampons et deux FPGA (XC 3190) pour construire les cellules ATM et exécuter les protocoles AAL 5 et ATM.

Le DAN s'inspire du projet Pandora [Hop90] développé en collaboration entre Olivetti Research et l'Université de Cambridge. Pandora est orienté sur la vidéo et l'audio [Jon93]. La machine définie est très complexe. Elle contient la carte d'interface au réseau, la carte disque SCSI, la carte d'acquisition vidéo avec compression, la carte de restitution des sons, la carte pour la restitution de la vidéo (décompression, mixage pour incrustation sur l'écran de la station), et la station de travail. Chaque carte possède un Transputer.

Au niveau logiciel, une architecture supportant plusieurs services (MSNA : Multi-Service Network Architecture) a été définie [Mca89, Gmc92]. Elle s'inspire fortement du protocole ATM pour les couches basses (jusqu'au niveau AAL), mais elle est prévue pour utiliser plusieurs types de couches basses (Ethernet, Cambridge Fast Ring [Hop88], Fairisle, B-ISDN). Au niveau supérieur, la nécessité de séparer les flux de données et de contrôle ("in-band" versus "out-band") est mise en avant, ainsi que la garantie de qualité de service pour les applications temps réel. Un protocole de transport pour supporter les RPC est défini et offre un gain de 40% par rapport aux solutions classiques. Des mesures de performances de MSNA sont présentées dans [Gre92]. Elles montrent qu'un débit de 9 Mbit/s est obtenu pour un aller retour avec une carte ATM à 100 Mbit/s, utilisant un mode de transfert programmé. Avec un DMA, le débit s'élève à 61 Mbit/s pour la

transmission de messages de grande taille. Cependant, les limites de ces performances sont dues à la carte ATM utilisée plutôt qu'à l'exécution des protocoles de MSNA.

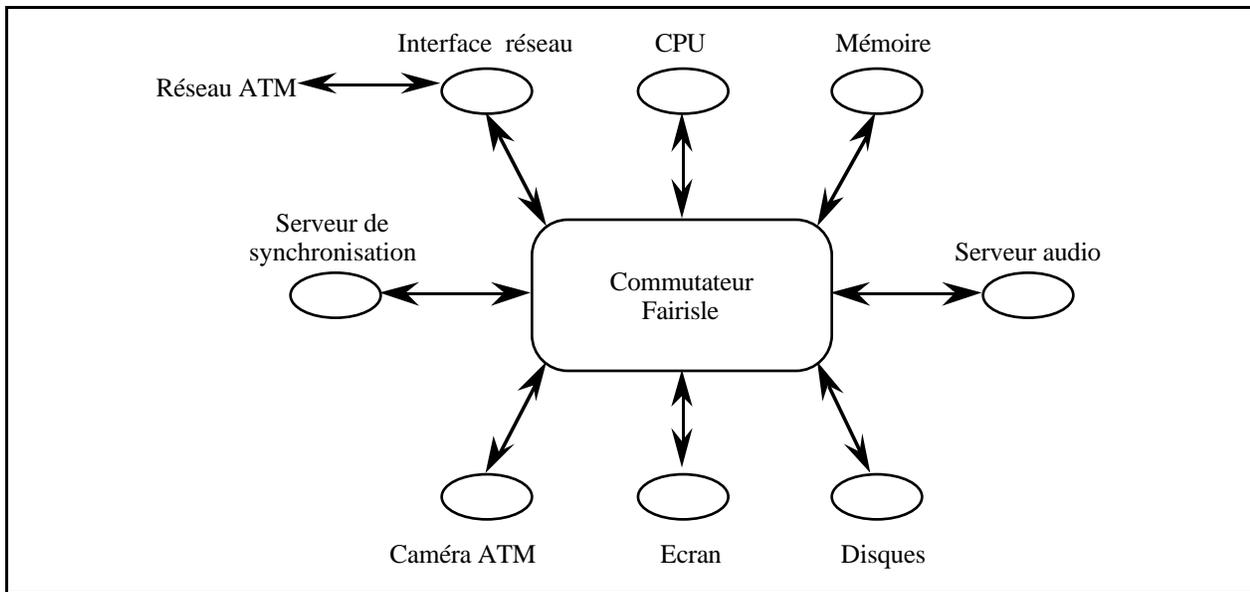


Figure II.1 : La station multimédia du DAN

Un système d'exploitation pour supporter les applications réparties est étudié dans le projet Pegasus [Les92]. Des extensions de ce système sont présentées dans [Mca93]. En particulier, une solution où les flux multimédia continus seraient directement écrits dans une zone réservée du cache est abordée. Elle est basée sur la constatation que l'audio et la vidéo sont généralement jouées dès leur arrivée sur la station. Elle permettrait de réduire le délai d'acheminement des flux vidéo, en évitant en particulier une copie entre la mémoire primaire et le cache. Ce système est en cours d'étude.

En résumé, l'Université de Cambridge propose un système multimédia complet et original où le bus fond de panier est remplacé par un réseau ATM. Cependant, la nécessité d'une carte d'interface ATM pour chaque composant du système conduit à une architecture très coûteuse. De plus, la séparation de la mémoire primaire et du cache conduit à des délais d'une dizaine de microsecondes. Cette architecture est adaptée aux flux audio et vidéo, mais pas aux autres applications temps réel.

II.3.1.2. Micronet Machines

La machine Micronet développée à L'Université Technique de Berlin [Pfe93] utilise le même concept que le DAN, c'est à dire remplacer le bus fond de panier de la machine par un réseau proche d'ATM. Cependant, le réseau n'est pas basé sur un commutateur mais sur un anneau. Les éléments (processeur, mémoire, caméra, ...) sont connectés à

l'anneau par des commutateurs réduits ("microswitch"), dont le but principal est le routage entre l'anneau ou l'équipement. Chaque élément possède deux commutateurs réduits puisque l'entrée et la sortie sont séparées sur des points d'accès au réseau distincts (cf. figure II.2). Le fait de séparer l'entrée de la sortie permet d'optimiser la bande passante disponible en fonction de l'application. Néanmoins, ceci n'est efficace que si les éléments sont placés correctement sur l'anneau en fonction des caractéristiques de chaque application. Par exemple, dans le cas d'une acquisition vidéo et de la restitution directe sur l'écran, il faudrait placer sur l'anneau le commutateur réduit associé à la sortie de l'interface de la caméra juste avant l'entrée de l'interface du dispositif de restitution.

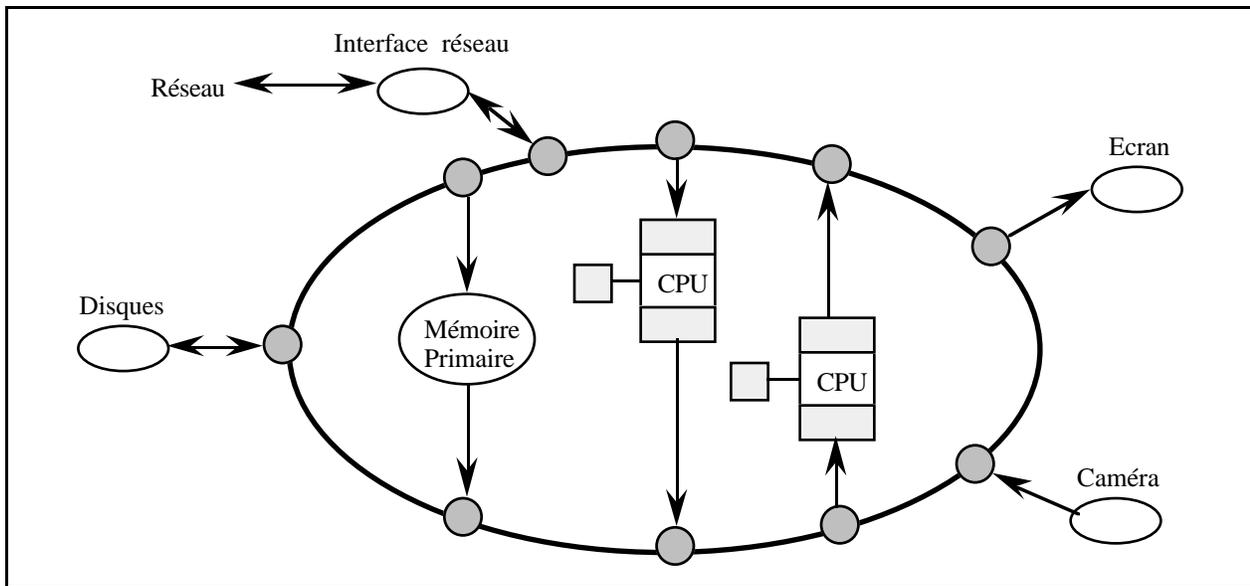


Figure II.2 : La machine Micronet

Pour les applications multimédia, un élément avec tampon d'entrée, processeur et tampon de sortie est utilisé. Les deux tampons, ainsi que la mémoire programme, sont indépendants afin d'augmenter les performances.

Cette machine permet de s'adapter facilement aux applications afin d'obtenir les performances maximales grâce au placement des éléments sur l'anneau. Cependant, pour des applications dont les transferts sont très disparates, il est difficile de prévoir les performances.

II.3.1.3. ATOMIC

ATOMIC [Coh93] est un réseau local haut débit construit par l'Université de Southern California / Information Sciences Institute. Il est basé sur des commutateurs, qui

peuvent être connectés à un autre commutateur ou à une interface hôte. Il est basé sur la technologie MOSAIC développée pour le traitement massivement parallèle à grain fin et à passage de message. Il permet d'obtenir un réseau local constitué de liens point à point à 500 Mbit/s [Coh92].

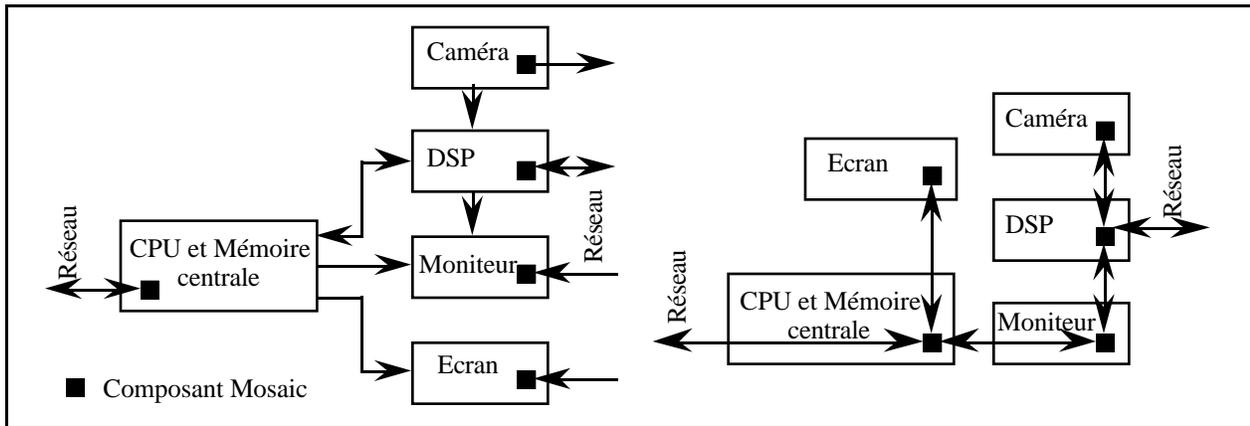


Figure II.3 : Les deux configurations possibles (Workstation et Netstation)

Sur ce réseau, une station de travail multimédia (Workstation) dispose de plusieurs accès, un par équipement (cf. figure II.3). Ainsi, le processeur, la caméra, le processeur de traitement du signal, l'écran de la station et le moniteur pour afficher la vidéo [Fin91] peuvent accéder directement aux autres éléments du réseau sans se préoccuper s'ils sont locaux ou externes. Une solution avec seulement deux accès au réseau est envisageable (Netstation) : un pour le processeur et un pour la partie vidéo. Chaque élément dispose d'un composant d'interface MOSAIC.

Ce réseau effectue un protocole minimum afin d'être efficace pour le traitement parallèle. Les performances obtenues ne sont donc pas comparables à celles des systèmes présentés ci-avant.

II.3.2. L'extraction des flux audio/vidéo au plus près du réseau

II.3.2.1. VuNet et ViewStation

L'architecture VuNet du MIT est constituée de trois éléments : des stations de travail, des équipements audio/vidéo et un réseau composé de commutateurs [Ada93]. Les équipements audio/vidéo (caméras, systèmes de traitement d'images, unités de disques, écrans) sont placés à l'extérieur de la station de travail et sont connectés directement au réseau (cf. figure II.4). La station accède aux équipements à travers le réseau et coordonne les flux d'informations. Cette architecture permet de partager les

équipements vidéo et de réduire la charge des tâches multimédia au niveau de la station. Un protocole similaire à ATM est exécuté au niveau de chaque équipement.

La station de travail ViewStation [Lin94] supporte le système VuSystem [Lid94] et est intégré au système VuNet. Contrairement aux autres systèmes, le traitement de la vidéo est réalisé par logiciel, pour que les applications puissent utiliser les flux vidéo. Avec une infrastructure de réseau supportant des débits de 500 Mbit/s, le transfert de flux vidéo entre le réseau et l'application, puis l'écran est de 35 Mbit/s [Ada94].

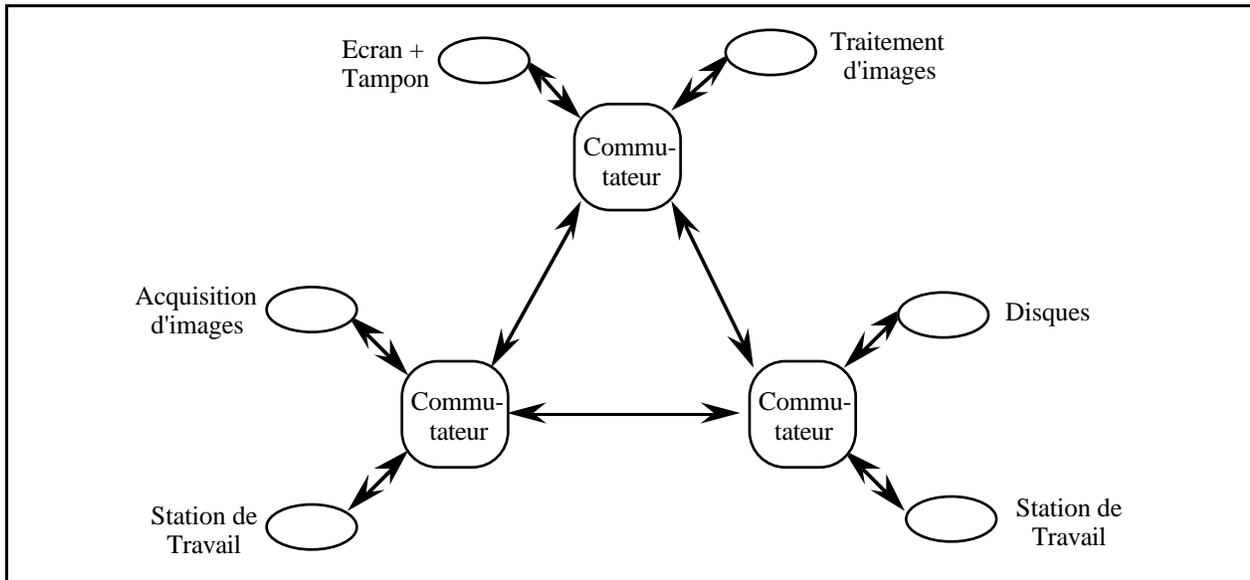


Figure II.4 : Le système VuNet

Cette solution prend en compte uniquement le traitement de l'audio et de la vidéo, sans se préoccuper des autres types de flux qui pourraient profiter des hauts débits. Ceci est compréhensible puisque les applications étudiées dans ce projet sont la surveillance d'une pièce, le tableau blanc, la caméra fouineuse (robot mobile intégrant une caméra) ou la diffusion d'un journal télévisé. Néanmoins, le débit disponible pour les flux vidéo reste assez faible, en raison d'un traitement logiciel important.

II.3.2.2. Washington University multimedia workstation

Une architecture moins ambitieuse a été développée à l'Université de Washington [Ric93]. Son originalité réside dans la séparation du traitement des flux audio/vidéo, ces derniers étant extraits (respectivement insérés) au niveau ATM et envoyés vers (respectivement reçus par) les cartes d'interface correspondantes, sans subir le traitement de l'AAL (cf. figure II.5). Les autres flux sont transmis au processeur hôte via des FIFOs. L'AAL 5 est exécutée sur le processeur embarqué (MC68030) et est utilisée pour les

données uniquement. Un CRC sur 16 bits est utilisé car le calcul du CRC 32 de l'AAL 5 est beaucoup trop coûteux en temps.

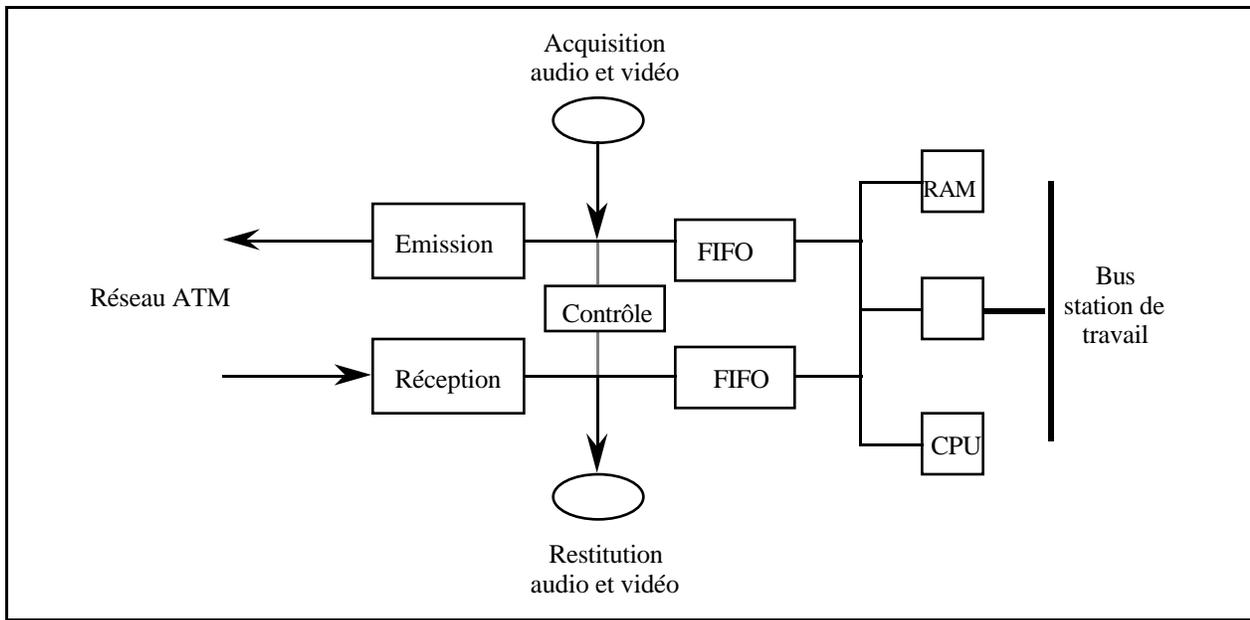


Figure II.5 : La station multimédia de l'Université de Washington

Ce système apporte une solution intéressante pour le traitement des flux audio/vidéo. Les cartes son et vidéo spécifiques permettent une restitution directe des signaux. Le système est bien adapté pour l'application visée, le transfert d'un flux d'images médicales. Par contre, l'ensemble des autres flux est traité de manière classique, avec une AAL en logiciel, et un nombre important de copies mémoire. L'intérêt de cette architecture est que les données audio et vidéo ne passent pas par le bus fond de panier.

II.3.2.3. Le système de l'Université de Lancaster

Le système développé à l'Université de Lancaster vise à intégrer les flux continus dans les systèmes distribués, en insistant sur les problèmes de synchronisation, de diffusion et de qualité de services à tous les niveaux [Bla94]. L'architecture de l'interface MNI (Multimédia Network Interface) [Bla93] permet de relier à un réseau haut débit une station de travail ou un PC, des équipements d'acquisition et de restitution audio et vidéo, ainsi que des disques (cf. figure II.6). Elle est constituée de six Transputers :

- Deux se chargent du protocole de transport. De plus, l'un d'eux gère les fonctions de synchronisation des flux continus (orchestration) [Cam92].
- Un effectue l'interface avec la station de travail ou le PC.
- Un est associé à des DSP pour l'audio.
- Un est associé à des DSP pour la vidéo.

- Un émule le réseau sous-jacent. Le but est de pouvoir émuler le réseau haut débit sous-jacent (ATM, FDDI et DQDB).

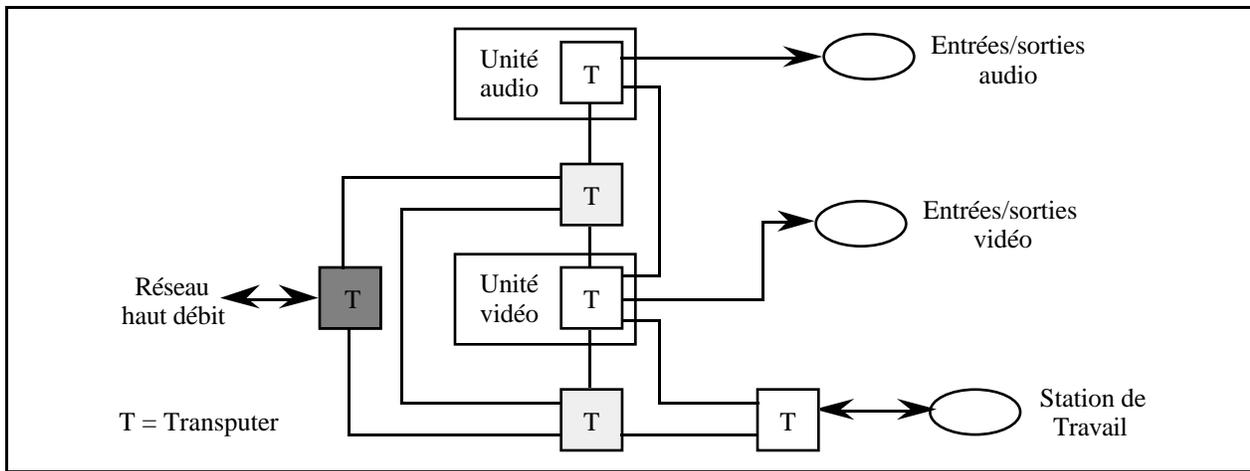


Figure II.6 : Le système de l'Université de Lancaster

Cette approche permet d'isoler la station des flux audio et vidéo mais les performances sont cependant faibles à cause des Transputers. En effet, les liens du Transputer permettent de communiquer à 20 Mbit/s. Le débit de bout en bout, mesuré au niveau transport, pour le transfert d'un flux vidéo est limité à 4,5 Mbit/s.

II.3.3. L'ajout d'un bus pour le traitement audio/vidéo

II.3.3.1. La station multimédia de NEC

L'architecture de la station multimédia est basée sur deux bus : le bus fond de panier et un bus multimédia [Sak93]. Sur le bus multimédia s'interconnectent les cartes d'entrée/sortie audio, d'acquisition vidéo, le processeur de traitement audio, les systèmes de compression vidéo, les disques, la mémoire et le contrôleur d'accès au tampon de trames vidéo (cf. figure II.7).

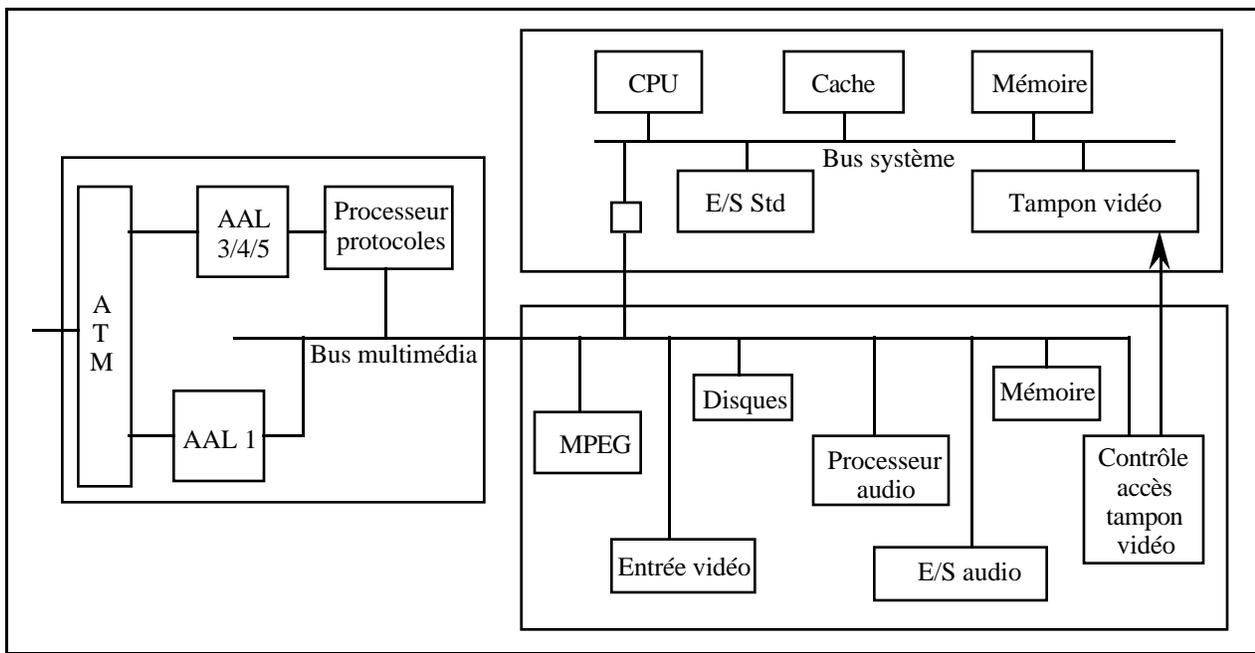


Figure II.7 : La station multimédia de NEC

Ce tampon est situé sur le bus fond de panier et sert à l'affichage sur l'écran de la station. L'interface au réseau ATM supporte tous les types d'AAL. Deux chemins de données sont utilisés, l'un pour les flux constants (AAL 1), l'autre pour les flux variables (AAL 2 à 5). L'interface est connectée sur le bus multimédia.

L'avantage de cette architecture est que l'ensemble de la station est assez compacte, et qu'elle traite tous les types de flux. Malheureusement, les performances de cette architecture ne sont pas connues.

II.3.3.2. Le système de l'Université de Melbourne

Dans l'architecture de la station de travail définie à l'Université de Melbourne [Arm90], trois types de flux sont identifiés : Vidéo/audio, Données et Signalisation. Les flux vidéo sont injectés (ou extraits) le plus près possible du réseau pour éviter la saturation du bus de la station. Pour identifier les différents flux, une allocation spéciale des VP/VC est utilisée.

Le système est constitué de quatre éléments : la station de travail, le répartiteur de flux, la carte vidéo et le réseau Ethernet (cf. figure II.8). Un réseau Ethernet a été choisi car ATM n'était pas encore disponible. Cependant, les cellules ATM sont encapsulées dans des trames Ethernet. La station de travail exécute le protocole AAL pour les données et la signalisation [Arm91]. La carte vidéo, indépendante de la station, permet de recevoir ou

d'émettre les flux vidéo. Le répartiteur de flux câblé [Lie92, Lie93] est chargé d'envoyer les paquets audio/vidéo vers la carte audio/vidéo et les autres vers le bus de la station. La liaison entre le répartiteur de flux et la carte audio/vidéo est faite par un réseau Ethernet. Le système nécessite donc deux réseaux Ethernet. Ainsi, pour communiquer entre la carte vidéo et le réseau Ethernet principal, il faut passer par trois interfaces Ethernet (sortie de la carte vidéo, entrée sur le répartiteur de flux, sortie de celui-ci).

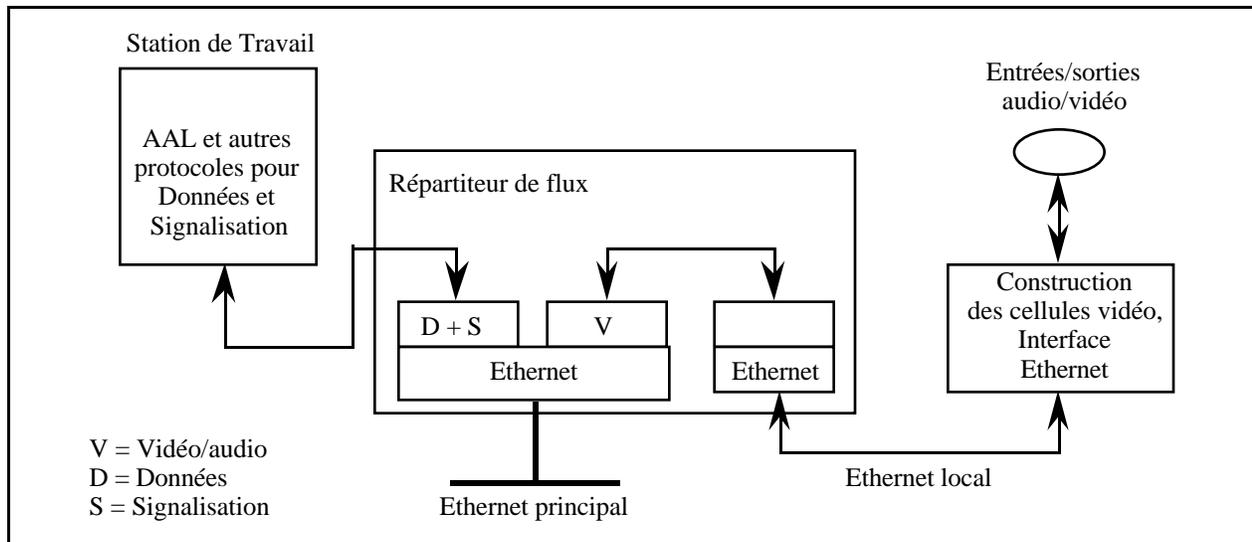


Figure II.8 : Le système de l'Université de Melbourne

Bien que l'idée de base soit intéressante, l'ensemble représente un système lourd, complexe et peu efficace. Il faudrait voir les résultats en utilisant un réseau ATM.

II.3.4. Résumé

Ce paragraphe a présenté des architectures variées permettant de traiter les flux haut débit, en particulier l'audio/vidéo. La table II.2 récapitule les caractéristiques des différentes solutions suivant les objectifs définis dans l'introduction. Elle montre clairement qu'il n'existe pas une solution permettant de répondre à tous les objectifs qui ont été fixés. En particulier, aucune n'aborde les problèmes de topologies de réseau ni de redondance. De plus, le traitement des flux temps réel se limite très souvent à l'audio et à la vidéo.

	Equipements Vidéo multiples	Topologies variées	Redondance	QoS pour applications temps réel	Télé-informatique classique	Coût
DAN	oui	indépendant	non	non	oui	très élevé
Micronet	oui	indépendant	non	non	?	très élevé
ATOMIC	?	limité	non	non	?	très élevé
VuNet	oui	indépendant	non	non	oui	élevé
Washington	oui	indépendant	non	non	oui	élevé
Lancaster	oui	indépendant	non	non	oui	élevé
NEC	oui	indépendant	non	non	oui	élevé
Melbourne	oui	indépendant	non	non	oui	élevé

Table II.2 : La comparaison des solutions suivant les caractéristiques recherchées

II.4. Conclusion

Ce second chapitre a fait l'état de l'art dans le domaine de l'architecture des machines de communication pour les hauts débits, en insistant sur les aspects multimédia. De nombreuses solutions ont été abordées. Pour les interfaces entre le réseau et la machine, la plupart des concepts pour obtenir des performances correctes est connue. Les différentes architectures de machines de communication multimédia présentées ont deux points communs :

- Elles nécessitent toutes des cartes audio et vidéo avec des interfaces particulières, intégrant les protocoles réseau de façon plus ou moins importante.
- Elles s'intéressent au traitement des flux audio et vidéo, mais laissent de côté les autres flux temps réel.

Dans le système que nous voulons définir, il est aussi important de supporter les flux audio/vidéo que le transfert de données temps réel. De plus, les problèmes de sûreté de fonctionnement doivent être pris en considération. Le chapitre suivant présente le système de communication répondant à ces besoins.

Chapitre III

L'architecture du système de communication proposé

Ce chapitre présente l'architecture du système de communication. La partie III.1 aborde les choix de conception matérielle et logicielle pour le couplage des équipements. La partie III.2 décrit les caractéristiques particulières des communications entre les équipements.

III.1. L'architecture de couplage des équipements

Les choix généraux qui influencent fortement l'architecture du coupleur sont d'abord présentés. Ils permettent de tracer les grandes lignes de la conception. Les détails de l'architecture matérielle sont ensuite introduits. L'organisation logicielle et le couplage entre ATM et les applications sont décrits dans la partie III.1.2.

III.1.1. Les caractéristiques matérielles

III.1.1.1. Présentation générale

III.1.1.1.1. Le coupleur ATM

Le coupleur ATM doit supporter des équipements aussi variés que des automates, des robots, des écrans, des calculateurs, des caméras, des microphones, des capteurs et des actionneurs. Aujourd'hui, la plupart de ces matériels sont présents dans les réseaux locaux industriels. Le plus important est d'intégrer la vidéo et l'audio de manière efficace, c'est à dire sans écrouler les performances du système. En effet, les flux vidéo demandent des bandes passantes élevées au cœur de la station et des traitements souvent lourds. Une implantation classique peut provoquer une saturation du bus fond de panier de la machine. D'autre part, les flux temps réel doivent profiter des améliorations introduites au niveau du réseau. Il faut cependant continuer à supporter les équipements actuels, sans avoir à redévelopper l'ensemble des interfaces et des logiciels de contrôle, au moins dans un premier temps.

Afin de répondre au besoin important de bande passante au cœur de la station (*caractéristique C1*) pour les applications audio/vidéo, deux solutions potentielles ont été envisagées :

- L'utilisation d'un bus local rapide dans la machine hôte. Deux bus normalisés émergent dans ce domaine : PCI [PCI93] et VL-Bus. Bien que ces bus offrent des débits élevés (132 Mo/s), transmettre plusieurs flux isochrones sur un bus partagé risque de conduire à des coûts de développement logiciel importants.

- L'utilisation d'un commutateur ATM interne à la station. Ceci répond bien au problème posé et permet en outre de construire des topologies variées à coûts réduits. En effet, plusieurs cartes d'une même station pourront échanger des données ATM sans passer par le bus fond de panier de la machine hôte. Un second bus haut débit (155 Mbit/s dans notre cas) est alors présent sur la machine.

Malgré les problèmes soulevés pour le bus local, la première solution permet d'obtenir une machine supportant les flux audio/vidéo avec une seule carte coupleur et des cartes d'acquisition ou de restitution du marché. Les deux solutions seront donc présentes dans la machine de communication. Le bus PCI a été retenu, puisque malgré sa jeunesse, il semble s'imposer dans plusieurs catégories de machines (PC, stations de travail, ...). Outre ces considérations pour l'audio/vidéo, l'utilisation d'un bus fond de panier standard est nécessaire, afin d'utiliser les cartes d'interfaces aux équipements industriels existants et les cartes processeurs classiques.

Afin de pouvoir offrir des configurations adaptées à chaque classe d'application, trois types de cartes ont été définies (cf. figure III.1) :

- a. Une carte d'adaptation ATM avec accès au réseau et processeur pour la gestion du réseau et le couplage à la machine hôte. Cette carte est la plus importante et doit être présente dans chaque station. Elle apporte l'intelligence au système de communication, et permet d'optimiser le transfert des flux temps réel.
- b. Une carte d'adaptation ATM pour les équipements audio/vidéo. Cette carte permet de supporter plusieurs caméras sur une même station, pour la télé-surveillance par exemple. Cependant, elle n'est pas obligatoire pour un ou deux flux vidéo puisque des équipements audio/vidéo peuvent se connecter sur le bus fond de panier PCI, et communiquer avec le réseau à travers la carte de type a. Elle permet de supporter directement la compression et la décompression câblée en temps réel des flux audio et vidéo. Elle pourra aussi traiter les AAL 1 et AAL 2 lorsque celles-ci seront normalisées.
- c. Une carte d'accès au réseau permettant d'avoir plusieurs connexions physiques au réseau pour le support de la redondance et pour la souplesse des topologies.

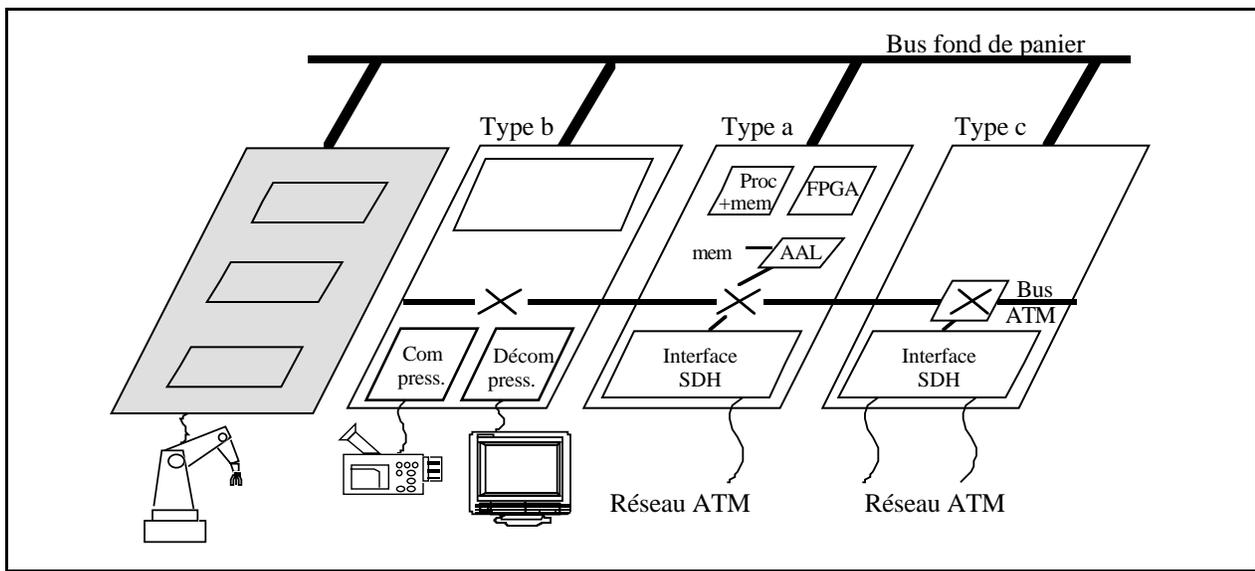


Figure III.1 : L'architecture générale du communicateur

Pour les cartes de type b et c, le bus fond de panier PCI sert uniquement à l'alimentation et à l'initialisation de la carte. C'est la carte de type a qui contrôle l'ensemble à l'intérieur d'une station. Une seule carte de type a est généralement présente dans une station.

Les cartes de type a et c permettent de construire un grand éventail de topologies donnant une large flexibilité au système. La première carte avec une ou plusieurs cartes du type b donnent à la station une grande capacité de transfert de flux audio et vidéo, grâce au bus local ATM interne de débit 155 Mbit/s. Le commutateur 4x4 (4 entrées et 4 sorties) présent sur chaque carte fait partie du bus ATM. Le paragraphe suivant montre la flexibilité du système défini en terme de configurations possibles.

III.1.1.1.2. Les topologies variées

En général, les réseaux ATM sont constitués de machines de commutation externes interconnectés par des liens séries point à point. Notre approche supprime ces commutateurs externes qui sont complexes et coûteux, d'autant plus que les solutions centralisées (en étoile) sont mal adaptées aux réseaux locaux industriels. En effet, elles requièrent des câblages longs et sont moins tolérantes aux fautes. En cas de panne d'un noeud de commutation, les stations connectées à ce noeud sont isolées du réseau et isolées entre elles. Au contraire, dans notre solution, lorsqu'une station tombe en panne, seule cette station est isolée du réseau. Il faut cependant mettre en oeuvre des mécanismes de reconfiguration. Notre solution permet d'obtenir un réseau compact composé uniquement de stations munies de cartes d'interface.

Les cartes de type a et de type c offrent un double accès au réseau afin de réaliser des topologies variées (*caractéristique C2*). La plus simple de ces topologies est un double anneau (cf. figure III.2). Dans ce cas, chaque machine a une carte du type a.

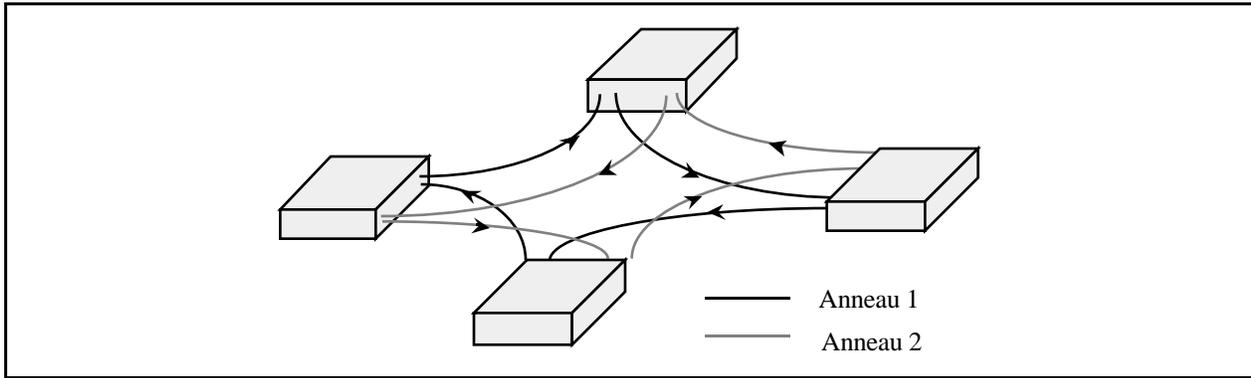


Figure III.2 : Topologie en double anneau

Avec seulement une carte de type a par machine, des topologies plus complexes peuvent être obtenues, par exemple en réduisant la dimension du réseau (nombre maximum de commutateurs traversés pour communiquer entre deux machines) (cf. figure III.3) ou en privilégiant certains liens de communication entre des stations (cf. figure III.4). Dans tous les cas, ces topologies restent tolérantes aux pannes (au moins pour la coupure d'un lien).

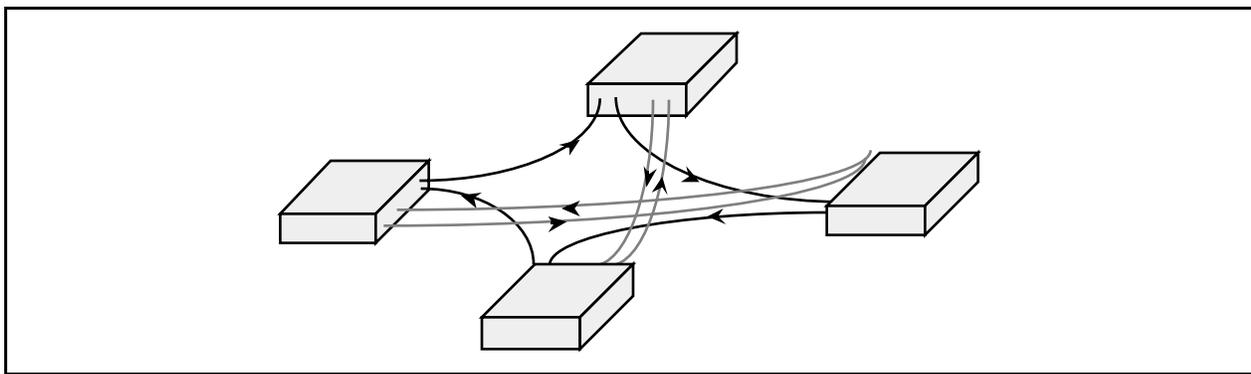


Figure III.3 : Réduction de la dimension du réseau

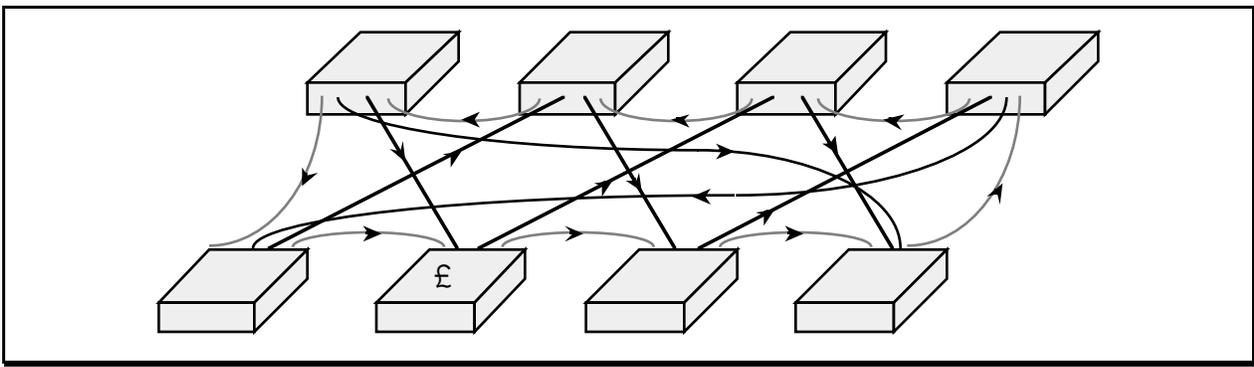


Figure III.4 : Topologie privilégiant certaines communications

Prenons un exemple concret : dans le cas de la télé-surveillance d'un tunnel ou d'une autoroute, les stations sont espacées le long de la voie. Avec une seule carte de type a par station, on obtient une structure en bus (cf. figure III.5). Cette configuration est une alternative efficace et économique à des commutateurs externes (cf. figure III.6). Dans notre solution, 2 câbles suffisent pour relier deux stations entre elles.

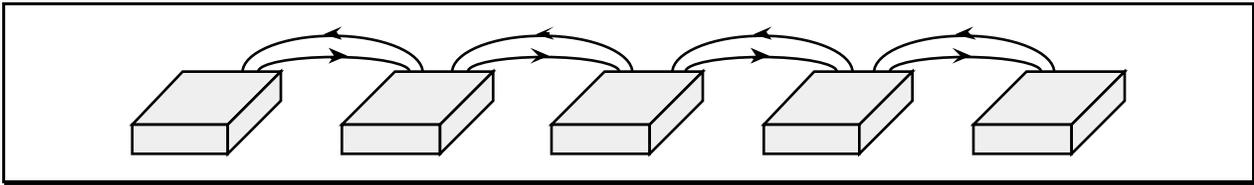


Figure III.5 : Structure en bus pour la télé-surveillance d'autoroutes ou de tunnels

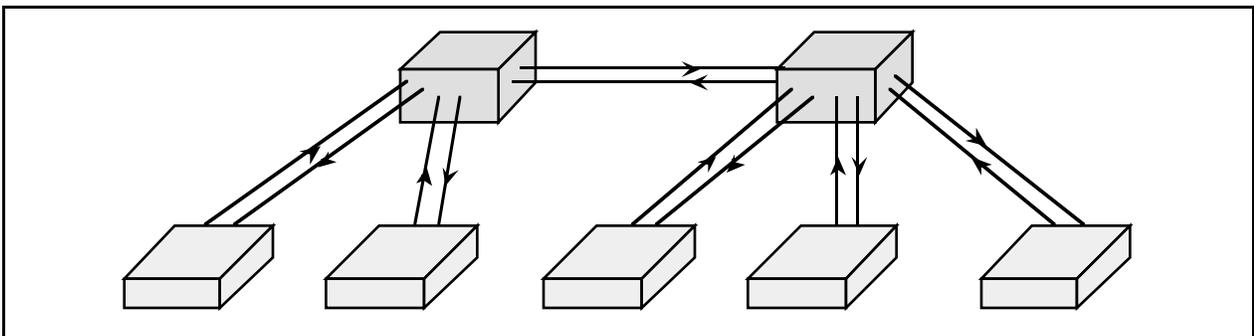


Figure III.6 : Configuration équivalente à la figure III.5 avec des commutateurs externes

Des topologies plus complexes peuvent être obtenues en ajoutant des cartes d'accès au réseau (type c) dans la station. La communication entre les cartes s'opère au travers du bus ATM. Ainsi, avec deux cartes dans une machine, nous disposons d'une machine de commutation 4x4 (cf. figure III.7). Les données traversant le commutateur 4x4 subissent deux commutations, mais ce n'est pas très grave puisque le délai de commutation est faible par rapport à ceux rencontrés dans les machines de commutation externe.

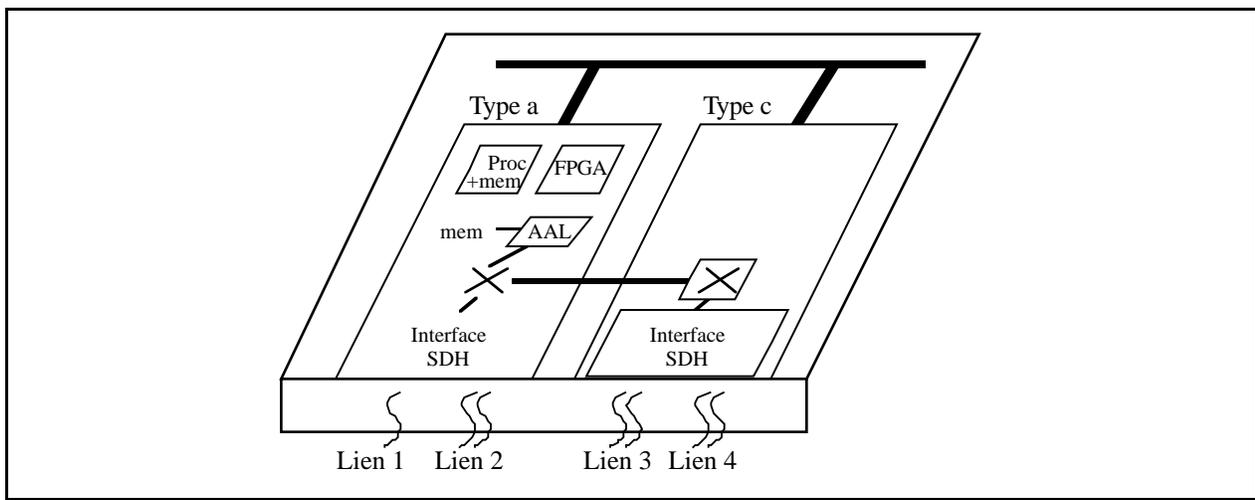


Figure III.7 : Configuration de la machine pour obtenir un commutateur 4x4

En plus de la construction de commutateurs, l'architecture définie permet de supporter la redondance de chemins physiques, indispensable dans la plupart des applications industrielles. En effet, nous avons vu qu'une carte de type a fournit une topologie en double anneau. Elle permet aussi d'assurer la redondance (cf. figure III.8) (*caractéristique C3*). Pour les topologies plus complexes, la carte de type c assure cette redondance.

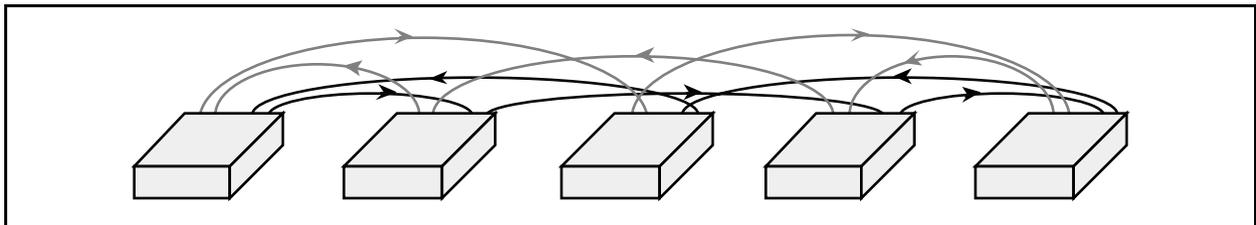


Figure III.8 : Configuration équivalente à la figure III.5 avec redondance

III.1.1.2. Les choix de conception

Le débit de 155Mbit/s choisi pose de nombreux problèmes pour la conception de l'architecture du coupleur. En particulier, il faut être capable de traiter 53 octets en moins de 2,7 us. Ceci passe par l'utilisation d'une logique câblée importante.

La carte la plus complète (figure III.1, type a) est à la fois une interface de communication pour la machine, un contrôleur de réseau et un routeur. Pour des raisons de modularité et de coûts de développement, cette carte est divisée en trois. Une carte mère d'adaptation à ATM supporte deux cartes filles, l'une pour l'interface physique au réseau ATM, l'autre pour le micro-contrôleur local. Les paragraphes suivants analysent successivement les caractéristiques des trois cartes, en justifiant les choix effectués pour l'architecture des différents éléments.

III.1.1.2.1. L'interface physique au réseau et la commutation intégrée

La première carte fille (cf. figure III.9) supporte l'interface physique au réseau ATM, ainsi que les éléments nécessaires à la commutation.

L'interface physique retenue est à la norme SONET/SDH 155 Mbit/s pour une transmission sur fibre optique. La hiérarchie digitale synchrone a été choisie par les opérateurs de télécommunication pour les réseaux publics. Il est paru naturel de choisir le même support physique pour notre réseau. La technique ATM a été au début prévue pour utiliser un support en fibre optique, ce dernier offrant des taux d'erreur très faibles. Un taux d'erreur inférieur à 10^{-9} est une des hypothèses fortes pour assurer le bon fonctionnement d'ATM. Depuis, des transmissions à 155 Mbit/s sur paire torsadée (UTP-5) ont été définies par l'ATM Forum.

Un ensemble composé du commutateur 4x4 et d'un traducteur d'en-tête ATM est au cœur du système de communication. Il effectue le routage des cellules vers la carte appropriée ou vers le réseau en fonction des champs VPI et VCI de l'en-tête et des tables de routage, ces tables étant mises à jour dynamiquement lors de l'ouverture et la fermeture des connexions, via le bus microprocesseur. Le commutateur embarqué a 4 ports bidirectionnels. L'un est utilisé pour le bus local ATM, un autre pour le transfert des données ATM vers la machine hôte et les deux derniers offrent un double accès au réseau afin de réaliser des topologies variées. Le port vers la machine hôte est connecté avec le composant de segmentation et réassemblage, et dispose d'une interface séparée du bus microprocesseur pour pouvoir supporter le débit de 155 Mbit/s bidirectionnel.

Afin de garantir les qualités de service d'une part et de limiter les coûts matériels d'autre part, un contrôle de flux crête est effectué sur chaque connexion en amont des commutateurs, ce qui permet entre autre de limiter les problèmes de congestion et de réduire la taille des FIFOs nécessaires au niveau des commutateurs. Les problèmes de congestion sont présentés dans la partie III.2.

Lors de la définition de l'architecture, un très petit nombre de composants permettant de réaliser un tel coupleur à un débit de 155 Mbit/s était disponible sur le marché. Un seul ensemble de composants (AAL [Fua93] + commutateur [Fub93, Fuc93] + interface physique [Fud93]) existait. En particulier, il existait un seul composant de commutation sur le marché permettant de réaliser ce coupleur. Nous avons donc dû adapter l'architecture aux caractéristiques des composants choisis. Les liens du commutateur ont un chemin de données de 8 bits. La fréquence est donc de 19.44 MHz, afin d'offrir un débit de 155,52 Mbit/s sur chaque lien unidirectionnel.

En particulier, le commutateur traite des cellules (53 octets) auxquelles ont été ajoutés 3 octets d'en-tête. Cet en-tête est utilisé pour router les cellules à l'intérieur et entre les commutateurs. Ainsi, il est nécessaire d'ajouter des convertisseurs entre le composant physique et le commutateur pour passer de 53 octets à 56 octets en réception et de 56 à 53 octets en émission. Pour cela, un translateur d'en-tête du même constructeur que le commutateur est utilisé. Il permet d'ajouter l'en-tête de routage aux cellules et de faire la translation de VPI/VCI si nécessaire.

Le composant d'interface physique de ce constructeur permet l'insertion et l'extraction de cellules par rapport à leur VPI/VCI ce qui satisfait les besoins d'administration (cf. partie III.2). Un convertisseur parallèle-série et série-parallèle et l'interface optique complètent cette carte pour le raccordement à la fibre optique. La carte disposant de deux accès au réseau, ces composants, ainsi que le translateur d'en-tête, sont présents en double exemplaire. Il est possible de changer ce bloc en fonction du type de transmission sur le réseau (cell based, SDH, ...) sans remettre en cause le reste de l'architecture.

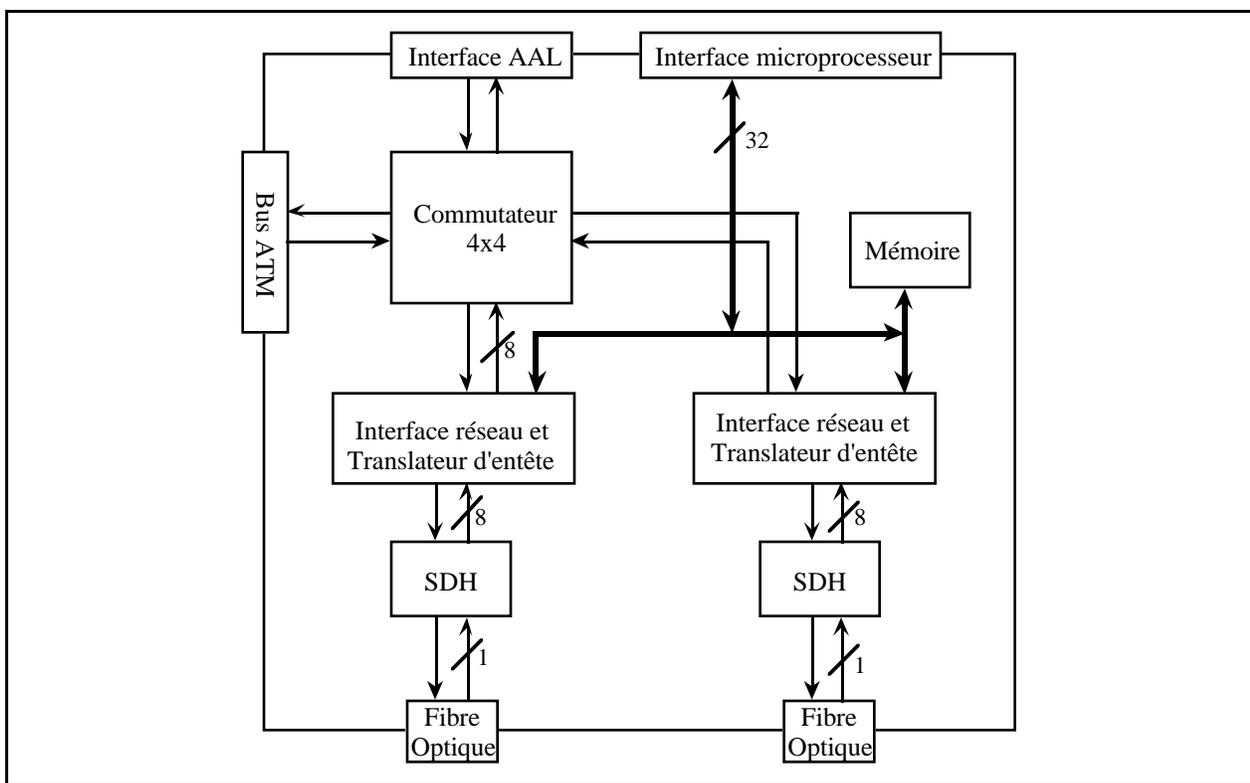


Figure III.9 : Schéma de la carte fille d'interface physique

Dans un commutateur, le stockage des cellules à aiguiller peut se faire de quatre manières [Tub91] :

1. Par des files d'attente en entrée.
2. Par des files d'attente en sortie.
3. Par des files d'attente en entrée et en sortie.
4. Par une mémoire globale partagée.

La solution 1 conduit à des performances faibles. En effet, lorsque plusieurs cellules sur des entrées différentes veulent aller vers la même sortie, une cellule dans une des files d'entrée désirent aller vers une sortie différente devra attendre. Les solutions 3 et 4 sont plus efficaces, mais plus complexes et plus coûteuses. La solution 2 est intermédiaire et offre de bonnes performances. Le commutateur choisi dispose de files d'attente en sortie (solution 2). Chaque file a une capacité de 75 cellules. La partie III.2 étudie les problèmes relatifs à cette taille de tampon, en particulier le contrôle de flux nécessaire pour assurer que les informations critiques ne subiront pas de perte et arriveront dans les délais.

Le paragraphe suivant présente la carte mère d'adaptation à ATM, support de la carte fille d'interface au réseau qui vient d'être décrite.

III.1.1.2.2. La carte mère d'adaptation à ATM

Après le choix des types d'AAL à supporter en fonction des services à offrir, le paragraphe s'intéresse à l'architecture et à la gestion des tampons de segmentation et de réassemblage.

III.1.1.2.2.1. Le choix du type d'AAL

Aujourd'hui, trois types de protocoles d'adaptation à ATM [Puj94] existent. L'AAL 1 a été définie pour les flux à fortes contraintes de synchronisation et à débit constant, par exemple la voix. Les AAL 3/4 et AAL 5 conviennent pour les données informatiques classiques, sans contrainte de synchronisation et à débit variable. L'AAL 5 a été définie a posteriori et est une version minimale de l'AAL 3/4. Contrairement à l'AAL 3/4, elle n'offre pas de niveau de multiplexage supplémentaire, elle opère un contrôle d'erreur sur l'ensemble du message et non sur chaque portion de 48 octets. De ce fait, il y a moins d'informations de contrôle, et par conséquent une surcharge beaucoup plus faible (8 octets par message pour l'AAL 5 contre 4 octets par message + 4 octets par portion de 48 octets pour l'AAL 3/4).

Pourquoi n'y-a-t-il pas d'AAL 2 ? Elle fut prévue à l'origine pour les flux à débit variable mais ayant des contraintes de synchronisation. C'est le cas par exemple pour l'audio et la vidéo compressée. Elle n'a pas encore été définie, et restera certainement dans

cet état. Aujourd'hui, l'ATM Forum travaille sur le transport de flux MPEG 2 (vidéo et audio compressée) [Mpe93] sur l'AAL 5.

Si l'AAL 1 est incontournable pour le transport du son non compressé, le transport de données de natures différentes nécessite une analyse plus poussée, qui doit notamment tenir compte des types d'applications visées. Si l'on prend en compte le type de données (classiques / isochrones) et le mode de transfert (assuré / non assuré), on peut identifier les quatre catégories suivantes :

- a. Les données audio et vidéo assurées : il s'agit par exemple du transfert d'une seule image (représentant un schéma, une carte, ...) ou du transfert d'une séquence vidéo courte (pour l'enregistrement ou le traitement d'images). Cette catégorie requiert un contrôle d'erreur (effectué dans l'AAL 3/4 et dans l'AAL 5) et une fonctionnalité de retransmission (acquittements et chiens de garde). Pour ce type de données, l'AAL 3/4 semble plus adaptée car elle peut permettre la retransmission de cellules et non de messages complets au niveau des couches supérieures. Il faut cependant développer un protocole de retransmission particulier. L'intérêt est que ce choix nécessite moins de mémoire (pour sauvegarder les cellules non encore acquittées). Il faut noter que pour retransmettre des cellules, il faudra être capable de les identifier, par exemple en utilisant un numéro de séquence sur 11 bits au lieu de 4 (en prenant le champ MID du niveau SAR).
- b. Les données multimédia non assurées : il s'agit par exemple de flux continus d'images (télé-surveillance). Cette catégorie ne requiert pas de retransmission mais nécessite simplement un contrôle d'erreur pour écarter les images corrompues. L'AAL 5 suffit donc dans ce cas. Néanmoins, la couche supérieure pourra être amenée à conserver en mémoire une certaine quantité d'informations afin d'assurer un flux continu.
- c. Les données classiques assurées : il s'agit par exemple du téléchargement, de la configuration des stations, de commandes (contrôle d'un robot) ou d'événements ainsi que des messages de signalisation. Cette catégorie nécessite, tout comme la première, la retransmission des données. Cette fois, la retransmission de messages complets est envisageable étant donné leur faible taille ou la phase concernée (initialisation). L'AAL 5 peut donc être utilisée dans ce cas, combinée à une fonction de retransmission.

- d. Les données classiques non assurées : il s'agit par exemple des messages en diffusion, des applications de téléinformatique et des remontées d'informations temps réel liées à la surveillance de variables (cas du rafraîchissement). L'AAL 5 est appropriée à cette catégorie puisqu'elle offre un bon rapport taille des données utiles / taille du message.

Suite à ce recensement des caractéristiques des différents flux, l'AAL 5 semble satisfaire tous les besoins (sauf pour le son non compressé, bien sûr). Seule la catégorie 'a' semble pouvoir tirer profit de l'AAL 3/4. Il faut cependant noter que pour les applications relevant de cette catégorie, et dans le cas d'un débit de 155 Mbit/s, la majeure partie des données sera émise sur le réseau avant que le destinataire reçoive la première cellule. L'argument de mémoire réduite avec une retransmission par cellule n'est donc plus vraiment valable. De plus, utiliser une retransmission qui oblige à violer la norme rend le système propriétaire et non interopérable.

Par conséquent, les AAL 1 et AAL 5 ont été retenues. Devant l'impossibilité de trouver des composants avec l'AAL 1 lors de la réalisation de cette étude, nous avons été contraints de ne garder que l'AAL 5 câblée. De plus, ce choix a été motivé par les avancées significatives dans le domaine de la compression de données en temps réel. Pour le son, le système se restreindra dans un premier temps au transport sous forme compressée.

Le composant AAL dispose d'un mécanisme de contrôle de flux par connexion, et d'un contrôle de flux sur l'ensemble des connexions. Il permet ainsi de réaliser le contrôle indispensable avant l'émission de données sur le réseau. L'utilisation de cette fonction pour garantir le délai d'acheminement des données temps réel est présentée dans la partie III.2.

Le paragraphe suivant présente un élément indispensable à la segmentation et au réassemblage : la gestion du tampon associé à l'AAL.

III.1.1.2.2.2. La gestion du tampon de segmentation et de réassemblage

Le composant AAL 5 choisi nécessite un tampon de segmentation et de réassemblage partagé avec le microprocesseur hôte. Il offre deux possibilités pour connecter ce tampon : soit utiliser une mémoire double port (DPRAM) soit utiliser une mémoire classique et passer par le bus microprocesseur (cf. figure III.10). Sur cette figure, l'accès du composant AAL aux fonctions ATM n'est pas représenté mais se fait par des liens dédiés distincts du bus microprocesseur. L'indication d'arrivée de données au niveau

de l'AAL se fait par interruption programmable sur le nombre de cellules ou sur le message complet.

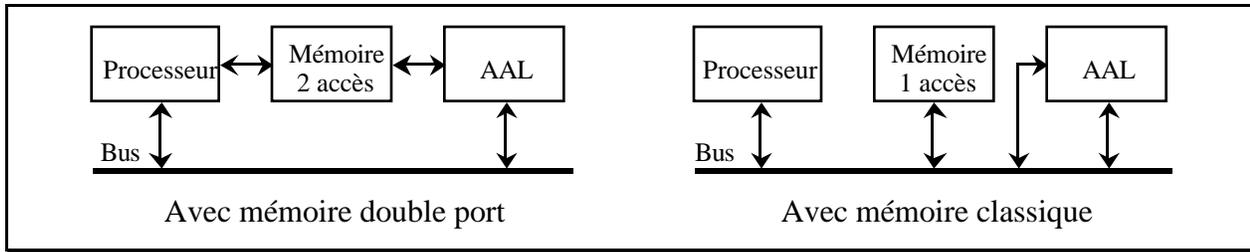


Figure III.10 : Les configurations possibles pour les accès au tampon

Pour que les applications puissent profiter pleinement de la puissance du réseau (155 Mbit/s en émission et en réception de manière continue), l'approche avec une mémoire double port a été choisie. En effet, à cette vitesse, la bande passante du bus sera en grande partie utilisée par le composant AAL. Pour transférer 155 Mbit/s simultanément dans chaque sens, il faut au minimum un débit sur le bus de 40 Mo/s, sans compter les informations de contrôle. Dans une solution avec une mémoire classique, les données nécessiteraient un débit de 80 Mo/s puisqu'il y a deux fois plus de transferts sur le bus. Ceci n'est guère envisageable avec les bus actuels.

Cependant, l'inconvénient de la mémoire double port est sa surface importante. Il conviendra donc de limiter sa capacité, ce qui peut avoir une incidence sur la capacité d'émission et de réception. Il est à noter que le composant AAL permet des accès à la mémoire en 32bits. Ainsi, tous les bus ont une largeur de 32 bits.

Afin de gérer les messages sur les connexions multiples, le composant AAL dispose de structures de données complexes dans la mémoire. Ces structures permettent la communication pour le partage des données entre l'AAL et le processeur. Elles servent d'une part pour la gestion du tampon de segmentation et de réassemblage et d'autre part pour l'indication d'arrivée de données sur une connexion en réception ou de disponibilité de données à émettre. La même mémoire contient à la fois les structures de données et les informations (cellules ou messages). L'allocation de la mémoire et la restitution des blocs est réalisée par le processeur.

Cette mémoire est gérée sous forme de listes chaînées de segments de messages de taille variable, ceci aussi bien en réception qu'en émission. Chaque segment possède un descripteur de 32 octets. Le nombre de descripteurs est limité à 4096. De plus, pour chaque connexion, un descripteur est associé au VPI/VCI et a une taille de 16 octets. Le nombre maximum de connexions simultanées est de 1024. La gestion de ces structures est

complexe, et similaire à celle d'un autre composant AAL existant [Tsw92]. Le paragraphe III.1.1.2.2.4. montre quelles sont les contraintes imposées par ces structures sur la taille de la mémoire. Regardons auparavant le problème de la délivrance des données.

III.1.1.2.2.3. Le mode de transfert (flot ou message)

Lorsqu'on utilise l'AAL5 (comme d'ailleurs l'AAL 3/4), deux modes de fonctionnement sont possibles : le mode flot ("streaming mode") ou le mode message ("message mode"). Dans le premier cas, l'AAL délivre les données au processeur par paquets d'une ou plusieurs cellules réassemblées, sans attendre que le message complet soit arrivé. Le nombre de cellules par paquet est programmable. En mode message, l'AAL attend que toutes les cellules du message soient arrivées pour restituer le message réassemblé au processeur. En émission, le processeur transmet les données dans le tampon d'émission soit par paquets de cellules (cas du mode flot) soit par messages complets (cas du mode message).

La première solution a l'avantage de demander moins de mémoire au niveau des tampons de segmentation et de réassemblage. Par contre, elle nécessite d'effectuer le réassemblage des messages à un niveau plus élevé dans la pile des protocoles. Ce mode de transfert est intéressant pour des machines intermédiaires sur le réseau qui nécessitent un traitement du protocole AAL, comme les serveurs de données sans connexion. Il évite de stocker l'ensemble du message avant de le réémettre. Pour un équipement terminal, son utilisation est discutable.

Dans notre cas, le mode flot a été choisi. L'utilisation de la mémoire double port en est la raison. En effet, ce type de mémoire a une surface importante pour une capacité réduite. Pour que le système fonctionne correctement, il aurait fallu une taille mémoire de quelques mégaoctets. Sachant qu'une mémoire double port de 128K x 8 bits a une taille d'environ 10 x 3 centimètres, il faudrait une carte complète pour disposer d'un mégaoctet de mémoire ! De plus, le coût serait prohibitif.

Dans ces conditions, il est nécessaire de disposer d'une seconde mémoire, de type classique et de taille plus importante, pour le tampon d'émission et de réception. Deux choix sont possibles :

- Utiliser la mémoire de la machine hôte. Ce choix imposera au processeur hôte d'effectuer la segmentation et le réassemblage, ainsi que les protocoles nécessaires entre l'AAL et les applications. Mais, dans le domaine visé (applications

distribuées temps réel), ceci peut avoir de graves conséquences sur les traitements temps réel.

- Ajouter une mémoire classique de taille importante sur la carte d'interface de communication. Cette solution oblige à opérer une copie de mémoire à mémoire supplémentaire entre le réseau et l'application. Par contre, avec l'ajout d'un processeur sur la carte (carte micro-contrôleur local), le processeur de la machine hôte est alors déchargé des tâches temps réel relatives aux protocoles réseau. Plusieurs composants supplémentaires sont nécessaires dans ce cas : processeur, mémoire.

Finalement, les deux choix sont conservés. La présence de la carte fille micro-contrôleur restera une option (cf. paragraphe III.1.1.2.4).

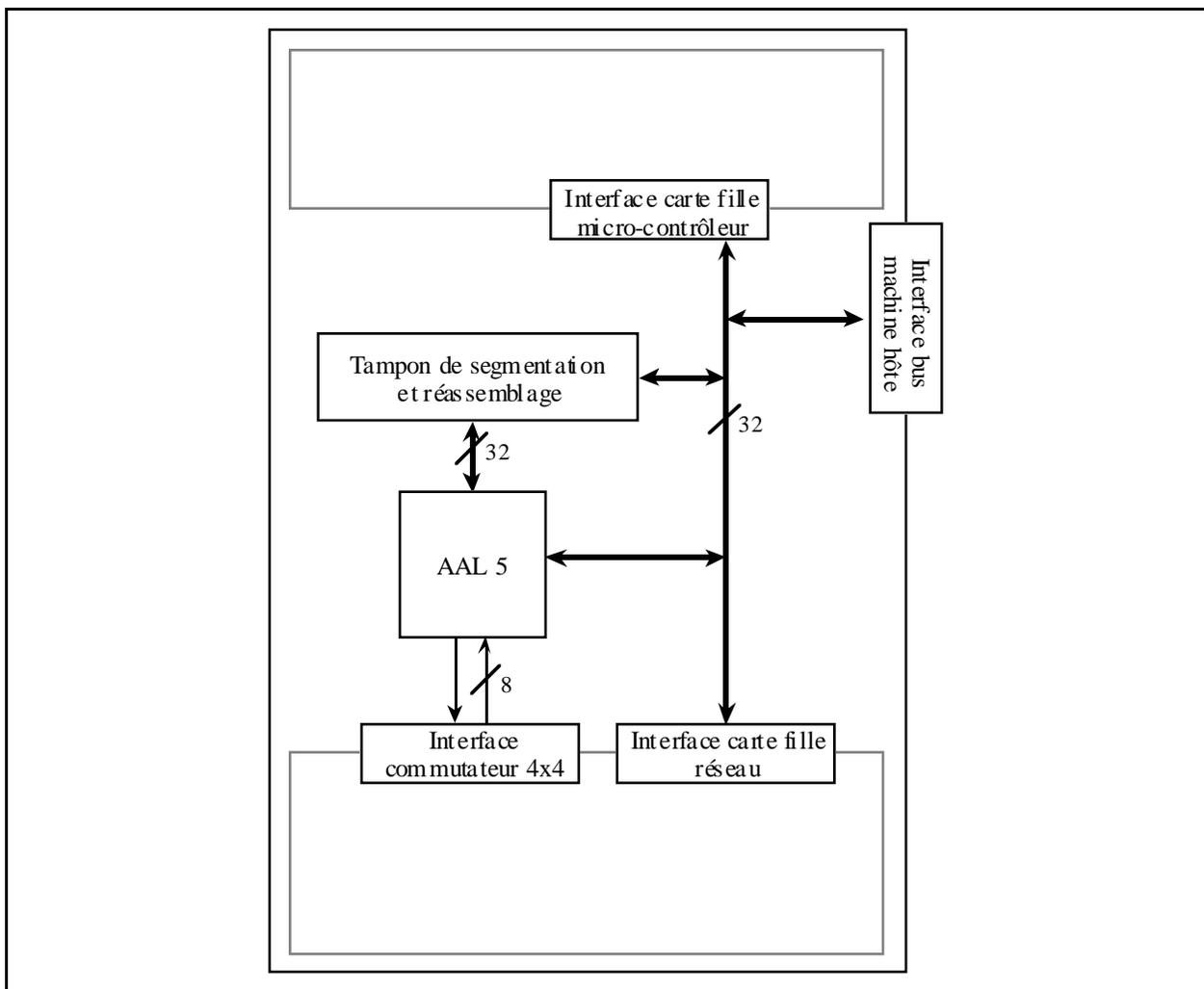


Figure III.11 : Schéma de la carte mère d'adaptation à ATM

L'architecture résultant des choix des paragraphes précédents et du présent est représentée sur la figure III.11. Le composant AAL et le tampon de segmentation et de réassemblage sont présents sur la carte mère. Le tampon d'émission et de réception est sur la carte fille processeur pour des raisons exposées dans le paragraphe III.1.1.2.4. La carte mère contient, outre les connecteurs pour les interfaces avec la carte fille de liaison au réseau et avec le commutateur, deux autres interfaces. L'une est indispensable au fonctionnement, c'est l'interface à la machine hôte. Elle est réalisée par un composant standard permettant d'accéder au bus PCI. Un autre connecteur permet d'ajouter la seconde carte fille contenant le micro-contrôleur local. La carte mère est assez simple pour permettre de l'adapter facilement à d'autres bus de machines hôtes (VME, S-Bus, EISA, ...).

Le paragraphe suivant étudie le dimensionnement des différents tampons.

III.1.1.2.2.4. La capacité des tampons

III.1.1.2.2.4.1. Le tampon de segmentation et de réassemblage

Nous avons vu que le tampon de segmentation et de réassemblage géré par l'AAL nécessitait des structures complexes. Ce paragraphe évalue la taille du tampon en fonction de ces structures, du nombre de connexions et de la taille des messages.

Supposons qu'il y ait en moyenne deux descripteurs par connexion. Les valeurs diffèrent peu si le nombre de descripteurs augmentent. La taille de la mémoire nécessaire est définie par la formule :

$$T = N * \text{desc_connexion} + N * 2 * \text{desc_segment} + C * \text{taille_cellule}$$

avec N : nombre de connexions actives (nombre de messages).

C : nombre de cellules par message (une cellule a 48 octets utiles).

T : taille de la mémoire (en octets).

desc_connexion : taille du descripteur de connexion (16 octets).

desc_segment : taille du descripteur de segment de message (32 octets).

taille_cellule : 48 octets.

En fonction de la taille de la mémoire et du nombre de messages, on obtient le tableau suivant :

T N	64 Ko		128 Ko		256 Ko	
	(1)	(2)	(1)	(2)	(1)	(2)
1024	impos	sible	1024	1,0	3755	3,7
512	512	1,0	1877	3,7	4608	9,0
256	939	3,7	2304	9,0	5035	19,7
128	1152	9,0	2517	19,7	5248	41,0
64	1259	19,7	2624	41,0	5355	83,7
32	1312	41,0	2677	83,7	5408	169,0

(1) : capacité en nombre de cellules.

(2) : nombre moyen de cellules par connexion.

Table III.1 : La capacité du tampon avec 2 descripteurs par connexion
(en nombre de cellules)

Seules des capacités mémoire de 64, 128 et 256 Ko sont prises en compte. Des mémoires plus importantes auraient une surface trop grande. A partir de la table III.1, nous voyons qu'une solution utilisant une mémoire de 64 Ko n'est guère envisageable. Elle interdit par exemple la solution avec 1024 connexions.

La solution avec 128 Ko semble acceptable. Pour 1024 connexions, la capacité représente une seule cellule par connexion. Mais dans ce cas, les débits sont en moyenne de 155 Kbit/s par connexion, même si leur répartition peut être très disparate. Certaines connexions à débit élevé auraient besoin de plusieurs cellules, alors que d'autres à débit faible auraient besoin d'une cellule de temps en temps. A ce débit, le remplissage et le vidage du tampon peut s'opérer. De plus, 1024 connexions est le cas limite supporté par le composant AAL utilisé.

Une mémoire de 256 Ko permettrait de laisser plus de souplesse au processeur pour remplir et vider le tampon. Néanmoins, sa taille plus importante est un handicap.

En conclusion, nous avons choisi la solution intermédiaire, soit une mémoire double port de 128 Ko.

III.1.1.2.2.4.2. Le tampon d'émission et de réception

Le tampon d'émission et de réception sert de relais temporaire pour les données à émettre ou en cours de réception. Il se situe soit dans la mémoire hôte, soit dans la carte-micro-contrôleur (cf. paragraphe III.1.1.2.4). Les structures nécessaires pour gérer le tampon d'émission et de réception sont similaires à celles du tampon de segmentation et de réassemblage, puisqu'il faut gérer les messages suivant les connexions. D'autre part, le processeur gérant le tampon d'émission et de réception doit connaître tous les paramètres

associés à chaque message pour pouvoir utiliser le tampon associé à l'AAL. Le tableau suivant donne le nombre moyen de message par connexion que peut contenir le tampon en fonction du nombre de connexions et de la taille du tampon.

	1 Mo	2 Mo	4 Mo	8 Mo
1024	944	1968	4016	8112
512	1968	4016	8112	16304
256	4016	8112	16304	32688
128	8112	16304	32688	65456
64	16304	32688	65456	130992
32	32688	65456	130992	262064

Table III.2 : La capacité du tampon d'émission et de réception
(longueur du message par connexion)

Le choix de la capacité mémoire dépend beaucoup de la taille des messages à transférer. Les messages temps réel des applications industrielles sont généralement courts. Par contre, les applications vidéo ou téléinformatiques utilisent des messages de taille importante (quelques kilo-octets). Les trois premières capacités de la table III.2 peuvent convenir.

La mémoire choisie a une capacité de 4 Mo, mais elle n'est disponible qu'en partie pour le tampon d'émission et de réception. L'autre partie est utilisée par des éléments présentés dans le paragraphe III.1.1.2.4.

III.1.1.2.3. *Le fonctionnement sans micro-contrôleur local*

L'architecture définie jusqu'alors permet d'obtenir une interface de communication ATM, c'est à dire de transférer les données entre le réseau et la machine hôte. Elle est basée sur le premier choix du paragraphe III.1.1.2.2.3. La configuration de la carte ATM est dans ce cas (cf. figure III.12) réduite à son minimum, c'est à dire qu'elle comprend la segmentation et le réassemblage (AAL), le tampon de segmentation et de réassemblage et l'interface physique au réseau. Le processeur de la machine hôte se charge des transferts entre la carte réseau et la mémoire de la machine hôte.

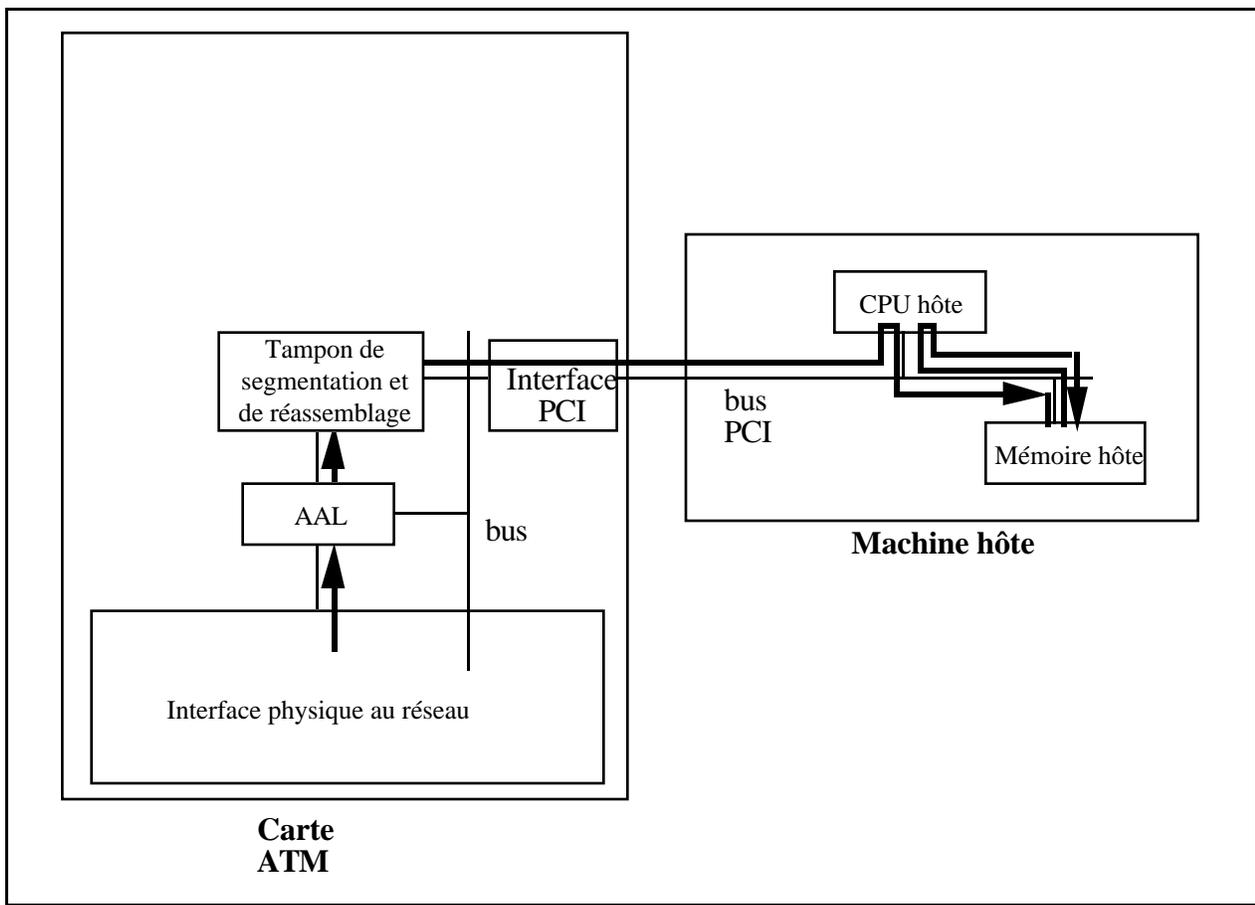


Figure III.12 : Les copies mémoire lors de la réception sans le micro-contrôleur local

Son fonctionnement est alors le suivant. Lors de la réception d'une cellule, celle-ci est tout d'abord stockée dans le tampon de réassemblage. Le processeur, sur interruption du composant AAL, transfère les données dans sa mémoire (tampon de réception). Il doit ensuite délivrer les données à l'application, en opérant une nouvelle copie, pour transformer les listes chaînées en un ensemble contigu. Les transferts sont alors en mode programmé ("Programmed I/O").

Ce mode de fonctionnement est une des utilisations de la carte de communication. Mais le système de communication offre des services supplémentaires avec la carte fille additionnelle présentée dans le paragraphe suivant.

III.1.1.2.4. Le micro-contrôleur local

Ce paragraphe définit l'architecture de la carte fille micro-contrôleur local qui permet de libérer le processeur des tâches relatives aux communications, et de faciliter le traitement des flux temps réel, et donc de répondre à la *caractéristique C4*. Une implantation des protocoles déportée sur la carte de communication n'est pas nouvelle

[Mic93]. Par contre, **l'introduction de primitives applicatives de haut niveau implantées au plus près du réseau est une idée novatrice pour faire profiter les applications des caractéristiques de la technologie ATM.** Tous les types de données ne peuvent être traités ainsi. Les classes de données transportées et les différents types de fonctionnements relatifs à ces classes sont définis dans ce paragraphe. Cette carte est une option du système, pour l'amélioration des performances temps réel.

III.1.1.2.4.1. L'architecture de la carte fille micro-contrôleur local

Cette seconde carte fille contient l'intelligence du système. Elle exécute un grand nombre de tâches relatives au protocole de communication ainsi qu'aux transferts entre le tampon de segmentation et de réassemblage et le tampon d'émission et de réception, et entre ce dernier et la machine hôte. Pour supporter ces tâches, la carte est composée de trois éléments principaux (cf. figure III.13) :

- Un processeur de type Transputer.
- Le tampon d'émission et de réception.
- Un composant programmable (FPGA).

et de deux éléments annexes :

- Une mémoire flash pour stocker un programme minimum pour initialiser le système à la mise sous tension.
- Une mémoire statique pour offrir une capacité de stockage au FPGA.

Le Transputer exécute les éléments logiciels relatifs au protocole ATM (protocole de transport, signalisation, gestion locale de couches). Il doit aussi gérer les structures de données utilisées par le composant AAL. Par rapport à d'autres processeurs, il possède l'avantage d'être un processeur simple à intégrer sur une carte externe, puisqu'il ne nécessite pas beaucoup de logique additionnelle. Il offre des performances intéressantes pour le traitement de données entières. Surtout, il ne nécessite pas de système d'exploitation, et apporte des garanties sur les contraintes temps réel, en particulier en terme de changement de contexte. Une des autres raisons du choix de ce processeur est l'expérience acquise dans ce domaine par notre partenaire industriel. Le modèle T425 à 30 MHz a été choisi. Il ne possède pas d'unité flottante puisque dans le protocole, les calculs s'effectuent sur des entiers. Une vitesse d'horloge assez élevée a été choisie afin de pouvoir répondre aux sollicitations du réseau, d'un débit de 155 Mbit/s dans chaque sens.

Le tampon d'émission et de réception est une Vidéo RAM qui stocke le code des programmes exécutés par le Transputer ainsi que les données à émettre ou en cours de réception. La VRAM dispose de deux accès, l'un classique, l'autre séquentiel fonctionnant en DMA. Ainsi, il est possible d'isoler les accès fait par le Transputer des accès avec le tampon de segmentation et de réassemblage.

Un composant programmable (FPGA : Field Programmable Gate Array) a été ajouté aux côtés du processeur. Ce type de composant peut jouer le rôle d'un co-processeur [Law94, Pot93]. Pour accélérer un programme, ces parties critiques en temps peuvent être programmées sur le FPGA alors que le reste du programme s'exécute sur le processeur. Le FPGA peut aussi être utilisé comme circuit spécifique pour des algorithmes qui utilisent des données de taille non standard, ou pour des opérations qui ne sont pas incluses dans le jeu d'instructions du processeur. Ces avantages peuvent être résumés par les quatre points suivants :

- Il permet un développement rapide puisqu'il n'y a pas de délai de fabrication.
- Il procure une grande flexibilité grâce à sa capacité de reconfiguration, en particulier pour le prototypage.
- Son architecture à parallélisme à grain fin permet d'obtenir des performances élevées.
- Son coût est faible puisqu'il est produit en grande quantité.

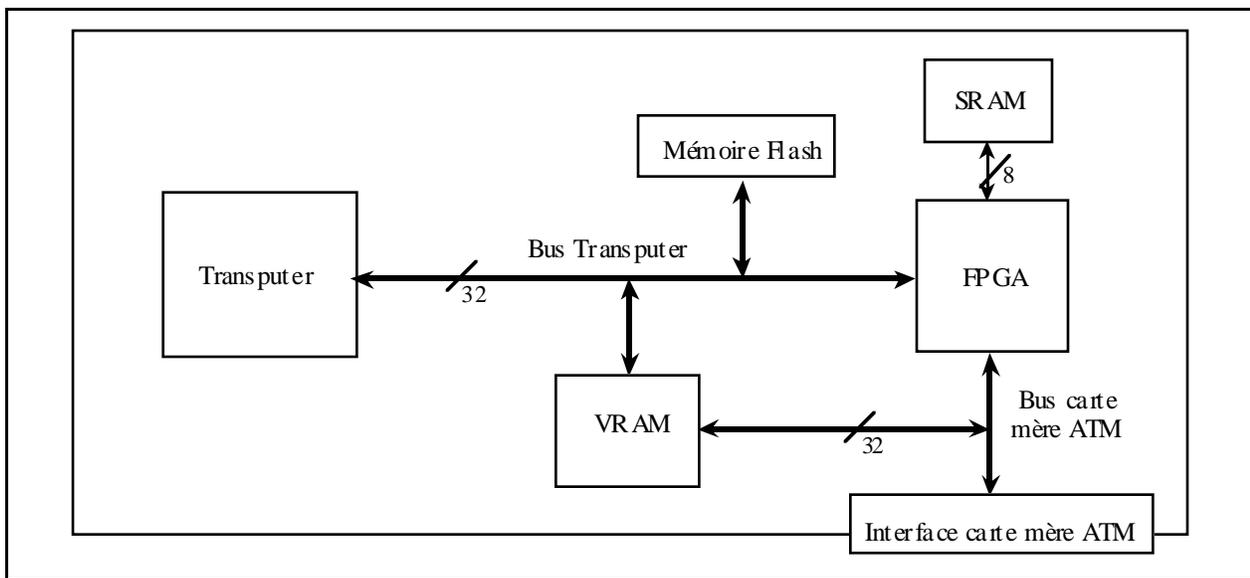


Figure III.13 : Schéma de la carte fille micro-contrôleur local

Le FPGA a plusieurs rôles. Le premier est l'adaptation entre les bus du Transputer et de la carte mère ATM. Le second est d'aider le Transputer dans les transferts entre les

différentes mémoires (tampon de segmentation et de réassemblage, tampon d'émission et réception, mémoire hôte). Il gère alors les transferts DMA avec la VRAM. Le troisième est de proposer des services distribués temps réel de haut niveau nécessitant un traitement rapide, pour assurer un délai d'acheminement de bout en bout réduit.

Le fonctionnement du Transputer est fortement couplé à celui du FPGA. La réalisation de cette carte a été renforcée par l'idée que des composants regroupant au sein d'un même circuit un processeur et un circuit programmable de type FPGA soient disponibles sur le marché dans un avenir proche.

Le paragraphe suivant définit le fonctionnement du système avec ce couple.

III.1.1.2.4.2. Le fonctionnement avec le micro-contrôleur local

Un des objectifs de la carte Transputer + FPGA est de minimiser le délai entre l'arrivée des informations dans le tampon de réception sur l'interface réseau et leur restitution à l'application. Pour l'émission, c'est le délai entre la demande par l'application d'envoyer des données et leur arrivée sur le réseau qui doit être minimisé. L'autre objectif est de pouvoir borner ce délai.

Dans ce cas, la carte fille micro-contrôleur local est ajoutée sur la carte de communication (cf. figure III.14). En réception, les données présentes dans le tampon de segmentation et de réassemblage sont copiées par paquets de cellules dans la VRAM, par le port séquentiel. Les traitements protocolaires (mode assuré, ...) à opérer sur les messages reçus sont effectués par le Transputer. Les messages sont alors délivrés à la machine hôte, sous forme d'un transfert DMA. Ce transfert DMA est géré par le FPGA. En émission, les transferts se font de la mémoire de la machine hôte à celle du Transputer puis vers la mémoire de l'AAL.

Par rapport à la configuration sans micro-contrôleur, le nombre de copies est identique. Par contre, le nombre de transferts sur le bus PCI est réduit au quart et le processeur hôte n'intervient que pour l'initialisation du transfert DMA. De plus, les protocoles sont traités sur le Transputer. Cette solution permet de mieux maîtriser les délais de communication, ce qui est l'objectif dans un système de communication temps réel. En effet, le processeur hôte est déchargé des tâches relatives aux communications.

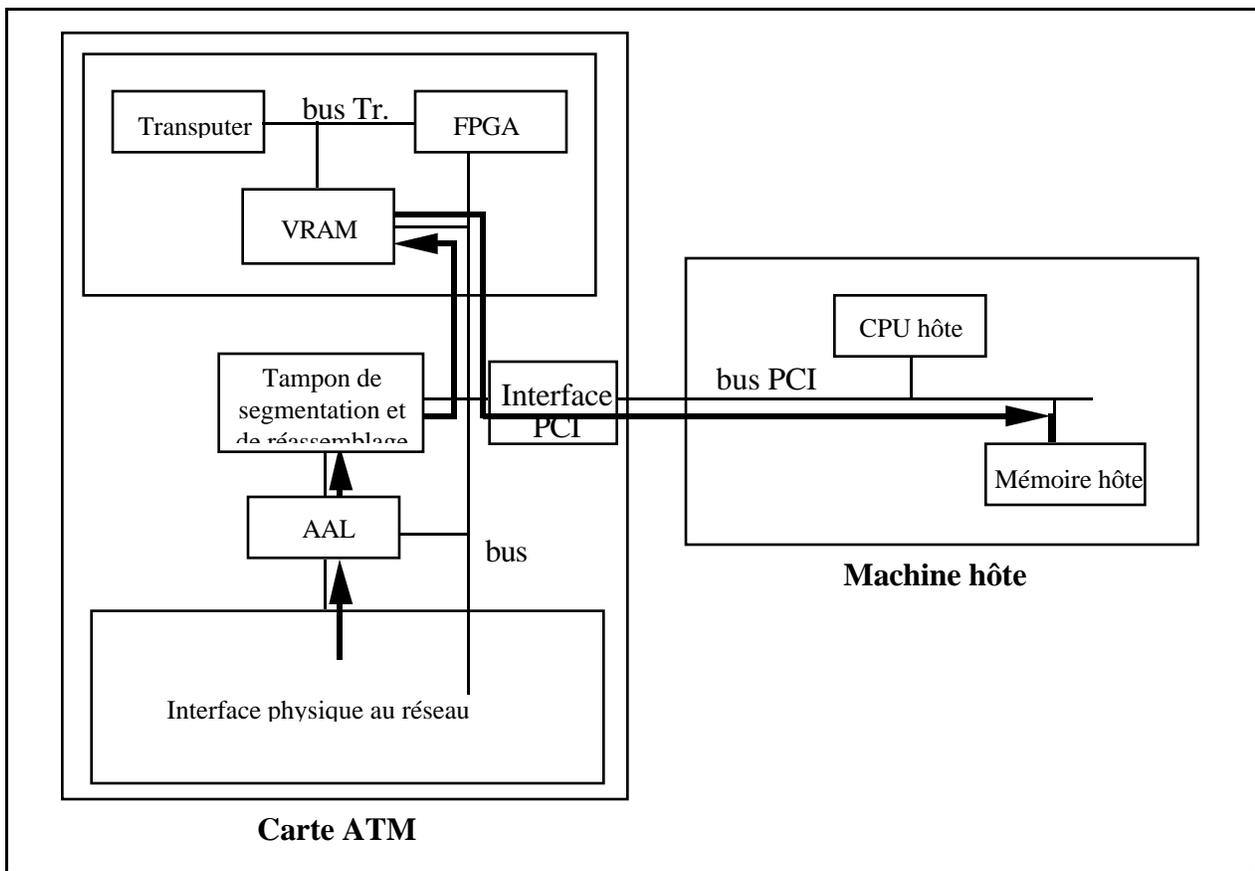


Figure III.14 : Les copies mémoire lors de la réception avec le micro-contrôleur local

Ce mode de fonctionnement convient bien pour les données classiques qui nécessitent des protocoles comme un transfert fiable. Par contre, certaines données temps réel n'ont pas besoin de subir des traitements protocolaires importants. Le paragraphe suivant isole un sous-ensemble d'applications qui peuvent tirer pleinement profit d'un traitement particulier. Pour chaque type de fonctionnement, il analyse la difficulté d'implantation et leur impact sur les débits que peut supporter le système.

III.1.1.2.4.3. Les types critiques de transfert d'information

Ce paragraphe montre quelles classes de données peuvent tirer profit des services directs au-dessus de l'AAL et définit leur fonctionnement.

III.1.1.2.4.3.1. Les flux à optimiser

Il n'est pas question ici d'optimiser l'ensemble du système mais seulement un certain nombre de primitives applicatives qui n'ont pas besoin de subir le traitement parfois très lourd de l'ensemble des couches protocolaires classiques. Il s'agit en particulier de la remontée périodique de variables de contrôle temps réel (comme défini dans FIP

[Tho93]) et des flux audio et vidéo. Pour ces derniers, il semble très intéressant de ne pas faire subir un traitement supplémentaire au flux de données entre la carte d'acquisition et l'accès au réseau (couche AAL), ou entre l'accès au réseau et la carte de restitution en réception.

Nous avons vu au début de ce chapitre que seuls les flux audio et vidéo compressés sont supportés dans un premier temps par le système. Le son non compressé disposera d'une AAL spécifique. Il existe aujourd'hui des standards comme MPEG [Gall91] qui permettent de coder le son et l'image simultanément. Ceci simplifie le problème de la synchronisation entre le son et l'image. La compression vidéo est basée sur la différence entre deux images consécutives. Seuls les changements sont transmis, et l'image entière n'est diffusée que de temps en temps (toutes les 15 images par exemple). Par conséquent, la perte d'une trame contenant une image complète peut être gênante pour la restitution, conduisant à brouiller toute une partie de l'écran. Par contre, une erreur de temps en temps sur un bit à l'intérieur d'une trame n'affecte pas la restitution, puisque des méthodes intrinsèques de récupération de ces erreurs sont prévues dans les codages.

Cependant, dans ces transmissions, la perte de données doit être évitée autant que possible, car la retransmission n'est guère envisageable. Si on veut pouvoir retransmettre une trame compressée corrompue, il faudrait disposer de tampons de très grande taille pour pouvoir stocker toutes les trames non acquittées. En effet, le débit de MPEG 1 [ISO11172] est de 1,5 Mbit/s et celui de MPEG 2 [Mpe93] peut atteindre 6 Mbit/s environ. Une trame MPEG 2 a une taille de 188 octets et nécessite donc un temps de transfert sur le réseau égal à 350 μ s. Si on ajoute les protocoles à chaque extrémité, le délai pour que l'émetteur reçoive l'acquittement est au minimum de 500 μ s, ceci dans le cas où il y a un acquittement par trame. Le débit nécessaire aux acquittements serait alors de 1,7 Mbit/s, ce qui n'est pas réaliste. Avec un acquittement pour 10 trames, il faudrait pour assurer une erreur de transmission un tampon au niveau de l'émetteur de 4 Ko. Mais, dans le cas d'une acquisition et d'une compression en temps réel des images et du son (c'est le cas dans la télé-surveillance), il faudrait stocker 4 Ko supplémentaires à chaque erreur, ce qui implique un tampon de taille non bornée. Avec des acquittements négatifs, il faudrait conserver tout le flux (il serait quand même possible de purger les plus vieilles trames après un certain temps).

Une autre solution est de stocker une certaine quantité d'images compressées au niveau du récepteur avant qu'il commence à retransmettre la vidéo. Mais cette quantité d'images dépend du nombre d'erreurs qu'il y aura lors de la transmission, et ce nombre est difficile à estimer.

En résumé, ces solutions sont viables seulement si le nombre d'erreurs de transmission est faible sur le réseau, ce qui est le cas avec ATM. Mais s'il y a très peu d'erreurs, pourquoi implanter un mécanisme de retransmission lourd et coûteux, qui ne

servira que très rarement. Il est préférable de gérer les connexions de manière à ce qu'il n'y ait pas de congestion, et donc de perte.

L'absence de retransmission pour les informations audio et vidéo compressées permet donc d'utiliser un protocole de transport plus léger.

Le problème de la transmission périodique de variables de contrôle temps réel est similaire. L'application type est le contrôle temps réel de processus à distance (relevés de température, positions d'automates, ...). Chaque processus de contrôle doit disposer localement d'une variable à jour. Une période de rafraîchissement P est définie pour un ensemble de variables. Aujourd'hui, les valeurs minimales de P sont de quelques centaines de millisecondes. L'objectif est d'arriver à une période de quelques millisecondes. Dans ce but, il est essentiel que le protocole de communication minimise le délai de transmission. La période étant courte, il n'est pas nécessaire d'effectuer de retransmission en cas d'erreur, puisque la valeur du cycle suivant arrivera avant qu'une retransmission n'ait pu être opérée. On peut voir le problème sous un autre angle : augmenter la quantité de variables transférées. Plus précisément, c'est le produit de la fréquence du cycle de rafraîchissement par la quantité de données qui doit être augmenté.

La remontée de variables sur changement de valeur d'une variable ou sur un événement (une combinaison de plusieurs variables dépassant un seuil) est plus difficile à optimiser. En effet, la détection du changement de la valeur ou du dépassement du seuil ne peut s'opérer qu'au niveau du processus. Il n'est pas possible de déporter ce calcul.

III.1.1.2.4.3.2. Le fonctionnement

Pour réduire le délai de transmission et de réception au niveau de la carte d'interface et du système d'exploitation de la machine hôte, le Transputer et le FPGA implantent un mécanisme permettant d'agir directement sur la mémoire locale des processus ou des équipements. L'intervention du système est alors réduite au minimum. Les deux paragraphes suivants décrivent le fonctionnement du mécanisme, respectivement pour la mise à jour de copies multiples et pour l'audio/vidéo.

a. La mise à jour de copies multiples

Le problème du rafraîchissement périodique de variables est similaire à un système producteur-consommateur distribué, où chaque consommateur doit disposer localement d'une variable à jour.

Dans le système producteur-consommateur classique, avec plusieurs producteurs et plusieurs consommateurs qui communiquent par un réseau, les producteurs mettent à disposition ce qu'ils ont produit et les consommateurs demandent à travers le réseau les variables qu'ils veulent consommer (cf. figure III.15). Ce système a un inconvénient : quand un consommateur a besoin d'une variable, il doit la demander et attendre son arrivée.

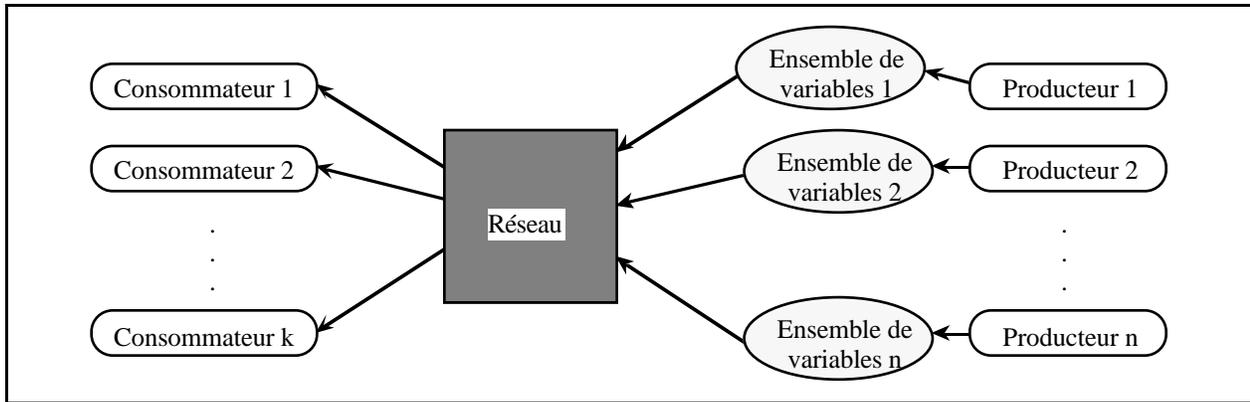


Figure III.15 : Le système Producteurs-Consommateurs classique

Dans le système envisagé, les producteurs envoient leurs données périodiquement vers les consommateurs qui en ont fait la demande (cf. figure III.16). La demande s'effectue une seule fois, à l'initialisation du processus consommateur. Les consommateurs disposent alors à chaque instant d'une copie à jour dans leur mémoire locale. Ce système peut être vu comme un ensemble de variables virtuellement partagées par l'ensemble des éléments du réseau (cf. figure III.17).

Le fonctionnement est ainsi complètement asynchrone, et découple le producteur et le consommateur. Chaque élément du système opère indépendamment des autres. L'application dispose alors localement à chaque instant de variables à jour, à la fréquence de rafraîchissement près.

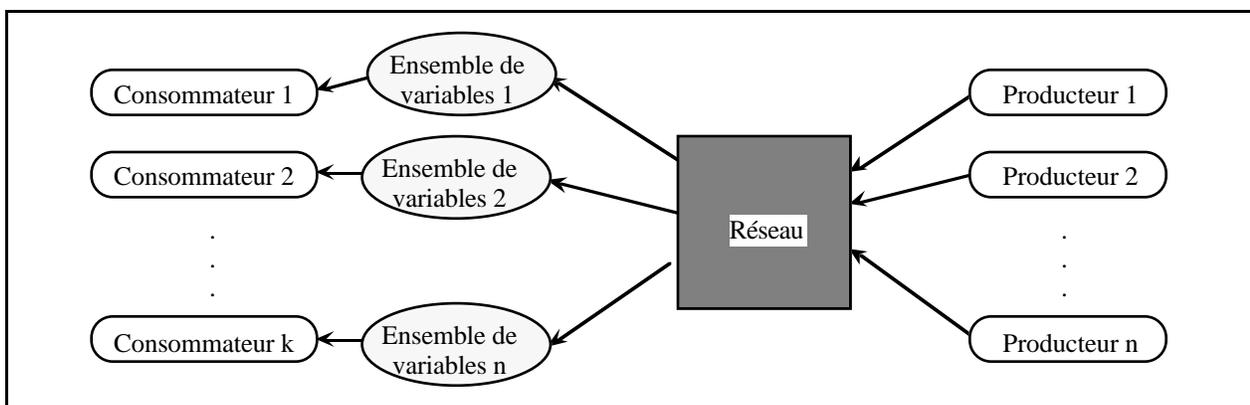


Figure III.16 : Le système Producteurs-Consommateurs envisagé

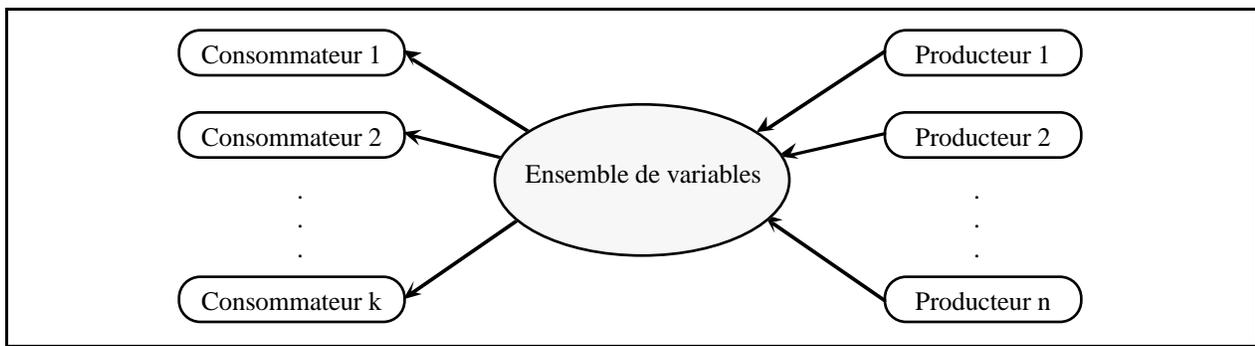


Figure III.17 : Le système Producteurs-Consommateurs virtuel

A chaque période de rafraîchissement, un producteur envoie plusieurs variables vers un ou plusieurs consommateurs. Un message contient donc plusieurs valeurs de variables. Pour distinguer les différentes valeurs, un nom unique est associé à chacune d'elles.

Les variables de contrôle temps réel sont stockées dans la mémoire locale du processus les générant (mémoire de la machine hôte) et dans les mémoires locales respectives de chaque processus les utilisant. Ce service peut profiter d'une implantation à bas niveau puisque les opérations sont effectuées sur des données de taille connue à l'avance. Le FPGA est naturellement adapté pour le réaliser. Du côté du producteur, un chien de garde déclenche le transfert des données de la mémoire locale au tampon de transmission AAL et construit le message à expédier. Du côté du consommateur, sur indication de l'AAL de la disponibilité de données, il décode le message et transfère les données dans l'espace mémoire locale. L'indication se fait pour une connexion (VPI/VCI) donnée.

Au niveau du producteur et du consommateur, le FPGA doit connaître la ou les connexions concernées, l'emplacement de la ou des variables en mémoire locale ainsi que les variables associées à chaque connexion. Il gère donc des structures de données contenant toutes les informations nécessaires à la gestion de ce service. Ces informations sont stockées dans la SRAM connexe au FPGA pour éviter à ce dernier d'accéder au bus à chaque variable décodée. La SRAM est initialisée par le Transputer.

Une évaluation des performances de ce service est présentée dans le chapitre IV.

b. La vidéo et l'audio

Pour la vidéo et l'audio compressée, le cas est similaire à la remontée de variables sur événements (données compressées prêtes à envoyer). Les transferts se font avec la

mémoire de la carte d'acquisition ou de restitution, lorsque compression et restitution se font en matériel sur la carte. Lorsque la décompression ou un traitement d'images est effectué sur le processeur hôte, il n'est pas utile d'utiliser le dispositif, puisque le temps de traitement est grand. Pour le transfert de vidéo compressée à partir d'un disque, on entre dans le cas du transfert de fichier. Nous nous intéressons uniquement au premier cas (traitement matériel). Cependant, dans certains cas, il ne s'agit pas d'une carte audio + vidéo, mais de deux cartes indépendantes. Il est alors nécessaire de séparer les flux audio et vidéo, par exemple dans le cas de MPEG 2. Il est ainsi possible d'utiliser le dispositif pour réaliser ce traitement.

L'implantation est similaire au service défini pour les variables de contrôle. La structure de données définie dans le paragraphe précédent est cependant à simplifier. En effet, une seule source de son et/ou d'images circule par connexion.

Pour les applications de télé-surveillance, il n'est pas possible d'afficher tous les écrans de contrôle vidéo simultanément. Le dispositif peut-être utilisé pour distribuer les différents flux vidéo les uns après les autres. Il n'y a alors qu'un seul flux vidéo qui circule sur le bus de la machine hôte. Il n'est donc pas nécessaire d'utiliser la carte de type b (cf. paragraphe III.1.1). De plus, l'opérateur peut sélectionner lui-même le flux. Il suffit de faire parvenir la commande à la carte de communication pour opérer le changement de sélection du flux.

Le dispositif doit aussi gérer le stockage des différents flux, et remplacer les anciennes données par les nouvelles fraîchement arrivées. Pour pouvoir changer de flux à n'importe quel moment, il est nécessaire que les flux stockés soient cadrés sur des unités d'information compréhensibles. Par exemple, pour un flux compressé en MPEG, il est nécessaire de commencer par une image complète.

III.1.1.2.4.3.3. Le concept

Ce dispositif se comporte comme un DMA auquel on donne à l'initialisation les adresses mémoires auxquelles il doit lire ou écrire les données. De plus, il utilise le réseau comme un DMA utilise le bus de la machine. C'est un DMA à deux adresses à travers le réseau. En effet, pour le transfert périodique de variables, l'application voit une copie directe du producteur au consommateur. Pour l'audio et la vidéo, c'est un transfert direct des capteurs d'image et de son (via la carte d'acquisition) à l'écran distant (via la carte de restitution). Le terme télé-DMA qualifie justement ce dispositif.

Dans ce système, les problèmes de configuration sont découplés de la transmission des informations temps réel. Un traitement minimal est effectué sur ces données afin de réduire le délai d'acheminement. C'est bien là la philosophie d'ATM.

III.1.1.3. Résumé

Ce paragraphe a présenté les caractéristiques matérielles de l'architecture de couplage des équipements. Les deux aspects importants de l'architecture matérielle de la machine sont :

- La présence d'un commutateur embarqué sur la carte de communication ATM qui permet de supporter plusieurs équipements audio/vidéo (*caractéristique C1*) sans passer par le bus fond de panier de la machine, d'offrir des topologies de réseau variées (*caractéristique C2*), et de proposer la redondance de chemins (*caractéristique C3*). Le tout permet de construire des réseaux compacts, sans machine de commutation externe.
- La présence d'un micro-contrôleur sur la carte de communication qui permet d'implanter au travers du couple Transputer + FPGA des services temps réel de haut niveau au plus près du réseau afin d'optimiser les transferts, par la notion de télé-DMA (*caractéristique C4*).

Pour que l'architecture de couplage des équipements soit complète, il reste à définir les éléments logiciels. Le paragraphe suivant présente leurs caractéristiques.

III.1.2. Les caractéristiques logicielles

La multitude de fonctions et de trafics à supporter implique la présence d'un grand nombre d'éléments logiciels. Le premier paragraphe discute l'organisation générale des protocoles et la répartition de ceux-ci entre le processeur embarqué et le processeur de la machine hôte. Le deuxième paragraphe s'intéresse aux différents services à offrir au-dessus de l'AAL pour supporter l'ensemble des flux et leurs diverses contraintes. Le troisième paragraphe décrit les protocoles supportés et le quatrième les interfaces offertes aux applications. Les fonctions globales du système de communication comme la signalisation ou l'administration sont décrites dans la partie III.2.

III.1.2.1. Présentation générale

Ce paragraphe présente l'organisation des logiciels et le choix d'implantation sur l'un des processeurs disponible dans la machine (processeur embarqué ou processeur hôte). Pour répondre aux besoins formulés précédemment, il faut supporter quatre catégories de protocoles :

1. Le transport d'informations de natures différentes implique plusieurs classes de services, offertes par la couche AAL d'adaptation aux applications, les couches physique et ATM étant indépendantes des types de services offerts. Les sous-couches SAR et CPCS de l'AAL permettent de transformer un message sous forme de cellules et vice versa et aussi d'effectuer un contrôle d'erreur par cellule ou par message suivant le type d'AAL. C'est dans la sous-couche haute de l'AAL (SSCS) que l'on trouve la gestion des acquittements et de la retransmission des cellules ou messages corrompus pour le mode de transfert assuré. Si on veut profiter des caractéristiques de l'AAL pour les couches supérieures, un protocole de transport rapide doit être implanté à ce niveau. En présence du micro-contrôleur local, ces fonctions sont traitées par le Transputer puisqu'elles font partie du protocole de communication.

2. Le protocole pour le traitement des flux temps réel et multimédia présenté dans le paragraphe précédent se décompose en deux parties :

- L'une s'occupe de l'initialisation et de la négociation des transferts (qualité de service). Elle est implantée sur le processeur embarqué.
- L'autre s'occupe des transferts d'information temps réel et se partage entre le processeur embarqué et le FPGA.

3. Le plan de signalisation, commun à plusieurs couches, a la charge de la gestion des connexions et, dans notre cas particulier, de la gestion des relations entre les identificateurs d'applications et les VPI/VCI correspondants de la couche ATM. Celui-ci s'occupe aussi de répondre aux demandes des autres entités de signalisation pour l'établissement de chemins sur le réseau ATM. Afin de ne pas ralentir les applications, ce protocole s'exécute sur le processeur embarqué. Son fonctionnement est décrit dans le paragraphe III.2.3.

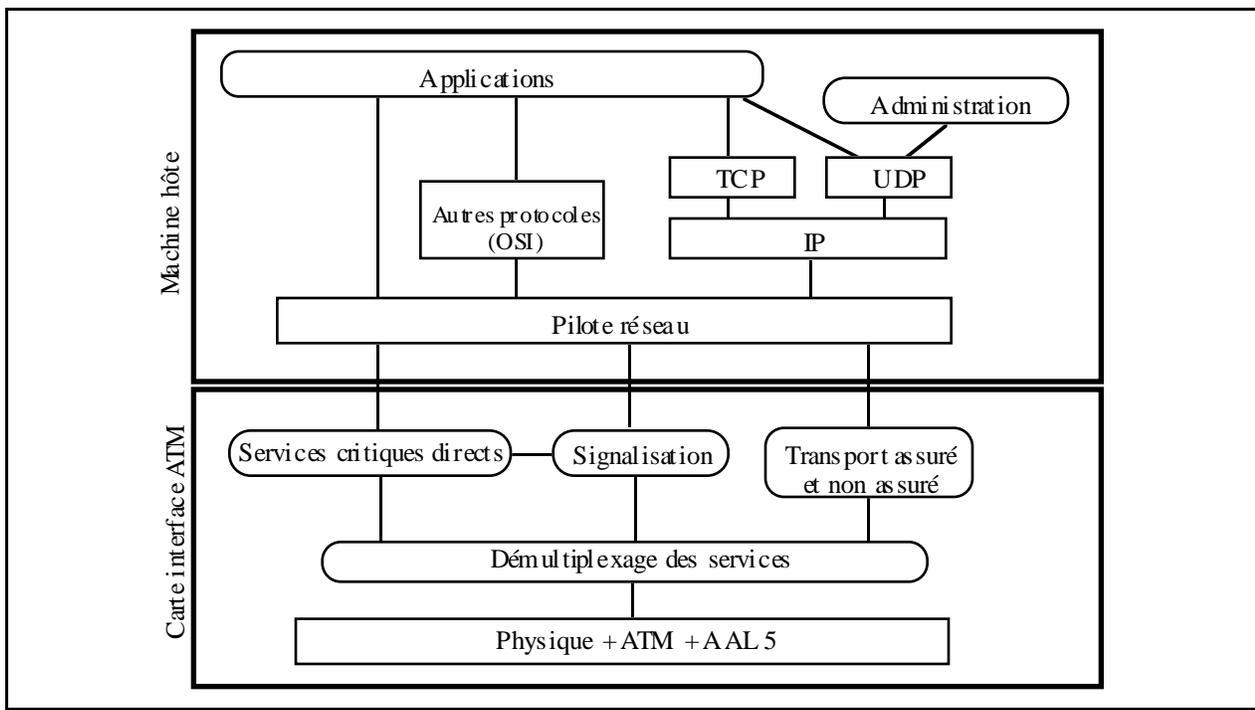


Figure III.18 : L'organisation générale des protocoles et des logiciels

4. Les protocoles TCP, UDP et IP sont présents dans le système pour assurer la compatibilité avec les applications existantes, et principalement les applications de téléinformatique classique (transfert de fichiers, terminal, ...) (*caractéristique C5*). Ces applications n'ont pas de contrainte temps réel. Par conséquent, une exécution de ces protocoles sur le processeur hôte a été retenue. Elle a l'avantage de permettre l'utilisation des suites TCP/IP existantes. Les protocoles OSI sont supportés de la même manière.

L'administration du réseau utilise le protocole normalisé SNMP (Simple Network Management Protocol) et les bases d'informations définies dans [RFC1695]. Elle utilise le protocole UDP pour communiquer.

Pour supporter ces différents services, plusieurs sous-couches SSCS au-dessus de la couche CPCSS doivent être définies. Elles nécessitent un démultiplexage des services (cf. figure III.18). Le paragraphe suivant présente les sous-couches supports et les services offerts.

III.1.2.2. Les choix de conception

III.1.2.2.1. Le couplage entre ATM et les applications

III.1.2.2.1.1. Les fonctions à ajouter à l'AAL pour offrir le service requis

De façon simplifiée, les AALs transforment les messages en cellules et vice versa. Celles-ci peuvent offrir des fonctionnalités différentes grâce à leur sous-couche SSCS. C'est en effet dans celle-ci que l'on trouvera des fonctionnalités spécifiques au mode assuré ou à la gestion des connexions. SSCS s'appuie sur le service de couche CPCS (cf. figure III.19). De plus, la couche SSCS est directement en relation avec les couches "non-ATM" supérieures (par exemple OSI 3/4 ou TCP/IP). A ce titre, la couche SSCS devra offrir les services attendus par la couche de restitution directe de l'information aux applications et offrir l'interface requise par OSI 3 ou IP. Ce paragraphe recense les modules SSCS nécessaires pour le support des protocoles présentés dans le paragraphe III.1.2.1. Il montre aussi comment l'AAL procède pour délivrer les messages reçus à la bonne entité SSCS.

La discussion des fonctionnalités nécessaires d'une couche SSCS (ou de plusieurs) est étroitement liée au choix du/des types d'AAL, ce choix étant lui-même lié aux types de trafic supportés et aux services offerts. Nous avons vu dans ce chapitre que l'AAL 5 a été retenue. Nous présenterons donc les fonctions à offrir au-dessus de l'AAL 5.

Nous avons vu dans les paragraphes précédents que les différents protocoles à supporter par le système peuvent être classés en quatre catégories. Chacune d'elles définit un type de sous-couche SSCS :

- Les applications directes au-dessus de l'AAL demandent un service de transport rapide.
- Les applications critiques pour lesquelles le transfert optimisé a été défini ont besoin d'un service SSCS particulier.
- La signalisation s'appuie sur un service AAL assuré, avec une couche SSCS normalisée par l'ITU-T dans [Q.2100, Q.2110 et Q.2130].
- D'autres protocoles, comme IP sur ATM ou le LAN Emulation se satisfont du service offert par la couche CPCS. Une couche SSCS nulle est donc présente.

Etant donné que plusieurs sous-couches SSCS sont présentes, il est nécessaire de démultiplexer les messages en réception pour les envoyer vers le protocole SSCS correspondant. Aucun paramètre n'est présent dans le CPCS-SDU pour indiquer la sous-couche SSCS. Par conséquent, le couplage ne peut se faire que lors de l'ouverture de la connexion. Il sert à informer l'entité appelée du protocole SSCS à utiliser pour traiter les

messages reçus sur la connexion ATM en cours d'ouverture. A cet effet, un champ de 8 bits (User Information) a été prévu dans le message d'ouverture d'une connexion (SETUP [Q.2931]). Aujourd'hui, les valeurs de ce champ ne sont pas normalisées. Seule la valeur 0 indique une sous-couche SSCS nulle. Nous avons donc choisi des valeurs arbitraires que nous changerons lorsque la norme sera définie.

Le processus de démultiplexage doit donc associer l'identificateur de la connexion avec cette valeur. Une table de correspondance est maintenue et mise à jour en étroite relation avec le processus de signalisation. Etant donné que nous travaillons uniquement avec l'AAL5 (pas de multiplexage sur les connexions AAL), l'identificateur de connexion de niveau AAL est le même que celui de niveau ATM. Ainsi, il est formé par la concaténation du VPI et du VCI.

Les différentes sous-couches SSCS nécessaires sont présentées sur la figure III.19. Une description détaillée de chacune d'elles est donnée dans les paragraphes suivants.

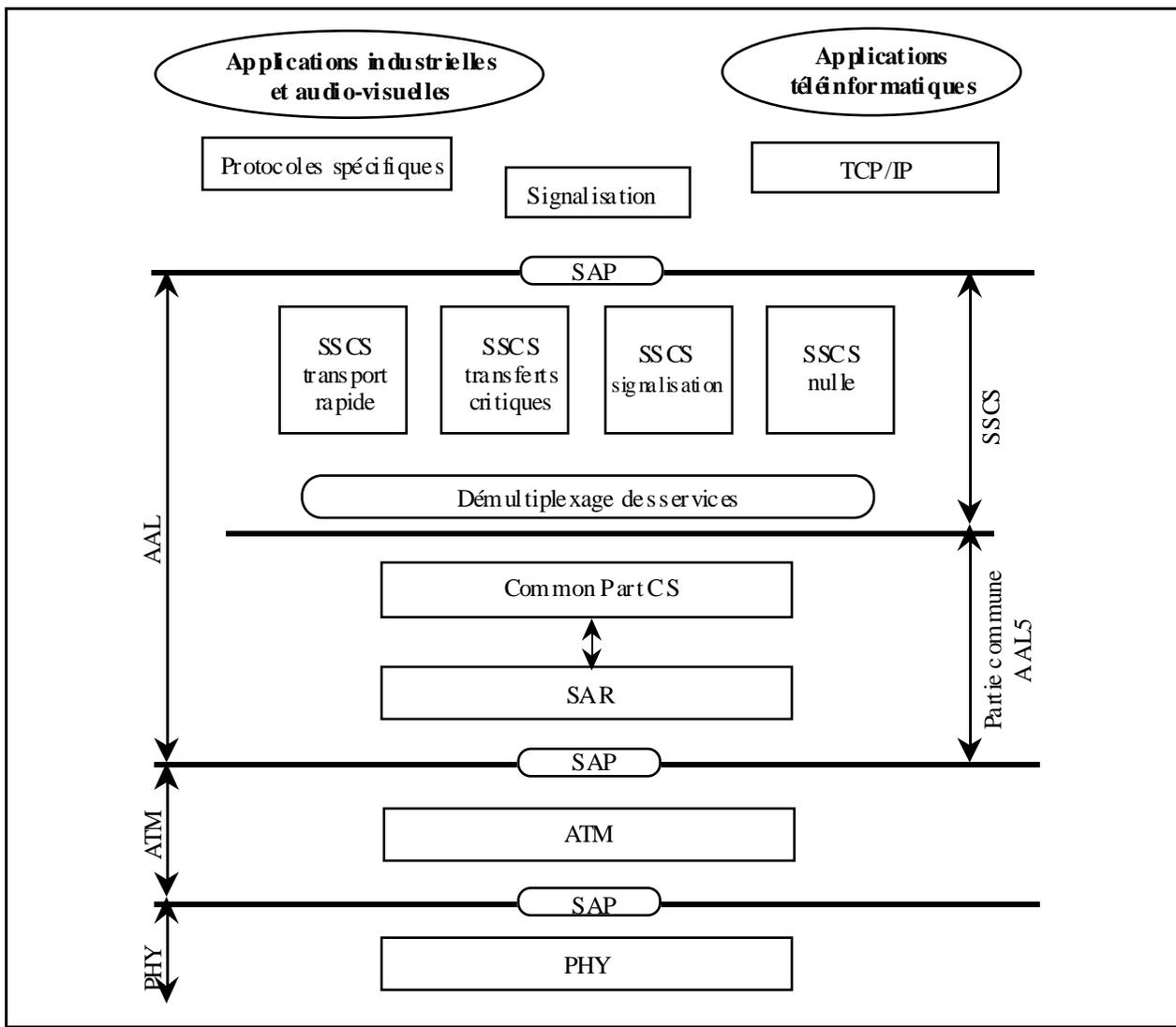


Figure III.19 : La structure des protocoles

III.1.2.2.1.2. SSCS transport rapide

Il est universellement accepté que les couches ATM et AAL comportent des fonctionnalités des couches liaison de données, réseau et transport du modèle OSI. Très peu de fonctions manquent pour offrir un service de transport.

Ce problème a été étudié pour la signalisation. La spécification du SSCS transport rapide s'inspire fortement de celle définie pour la signalisation ATM (SAAL : Signalling ATM Adaptation Layer) [Q.2100]. En effet, la norme précise que le transport des messages de signalisation doit se faire de manière assurée. L'entité réalisant le protocole du mode assuré est dénommée SSCOP (Service Specific Connection Oriented Protocol) [Q.2110] (cf. figure III.20). Par contre, le mode assuré pour les données n'est pas encore défini. Par conséquent, nous avons spécifié ce service en adaptant le protocole SSCOP.

III.1.2.2.1.2.1. Le protocole

Le mode assuré défini ici fonctionne par numérotation des messages envoyés et par retransmission en cas d'erreur. La taille maximum d'un message est de 65527 octets. Le mode assuré supporte les fonctions suivantes :

- L'intégrité du séquençement : cette fonction préserve l'ordre des segments de données.
- La correction des erreurs par retransmission : grâce à un mécanisme de séquençement, l'entité réceptrice détecte l'absence d'un segment de données. Cette fonction répare cette erreur en demandant la retransmission.
- Le contrôle de flux : cette fonction permet au récepteur de contrôler le débit auquel l'entité d'émission envoie les informations.
- Le rapport des erreurs à l'entité de gestion de couche (Layer Management).
- La récupération locale des données : cette fonction permet de récupérer des segments de données qui n'ont pas encore été émis ou acquittés.
- Le contrôle de la connexion : cette fonction établit, ferme et resynchronise une connexion en mode assuré.
- Le transfert de données : cette fonction est utilisée pour transférer des données en mode assuré (ou non assuré).
- La détection d'erreur sur le champ PCI (Protocol Control Information).
- Le statut : cette fonction permet d'échanger des informations de contrôle entre l'émetteur et le récepteur.

Les chiens de garde définis dans SSCOP sont statiques et non négociés. Ceci limite les performances du protocole puisqu'il est impossible d'estimer à priori le "round trip time" sur le réseau ou la charge des machines. Des algorithmes d'ajustement dynamique des chiens de garde ont donc été ajoutés.

Le protocole SSCOP a été défini pour offrir un service à la signalisation. Il ne prévoit pas de confirmer une donnée auprès de l'émetteur. Ceci est logique dans le cas de la signalisation puisqu'elle est essentiellement basé sur un mécanisme de questions/réponses, la réponse venant confirmer la question. Par contre, dans notre cas, d'autres applications utilisent le mode assuré, par exemple le transfert de fichiers ou d'autres applications classiques. L'option de confirmation auprès de l'émetteur a donc été ajoutée.

Pour le transfert des données classiques en mode assuré, on peut envisager de simplifier le protocole, en éliminant certaines options et en éliminant certains champs dans les paquets.

III.1.2.2.1.2.2. La qualité de service

Les applications s'adressent à la couche transport en lui spécifiant la qualité de service désirée. Le protocole se charge d'établir les connexions ATM et de négocier la qualité de service avec le processus de signalisation. Lorsque la connexion est ouverte, le transfert peut commencer. Il est important pour les applications de pouvoir renégocier la qualité de service en cours d'exécution.

Certains paramètres de qualité de service dépendent directement du réseau ATM, d'autres sont relatifs au service de transport. Aujourd'hui, ils ne sont pas encore normalisés dans ATM. Les paramètres de qualité de service que nous avons défini sont les suivants :

- Le débit.
- Le délai d'acheminement.
- Le transfert assuré ou non assuré.
- La priorité.
- Le taux de perte accepté.

Ce premier service permet d'offrir un transport rapide des informations, profitant des caractéristiques des protocoles ATM et AAL5 (code de détection d'erreurs calculé sur le message, ...).

III.1.2.2.1.3. SSCS transferts critiques

La couche SSCS relative au traitement optimisé des données temps réel a été définie pour pouvoir isoler plus facilement ces flux lors de la réception et ainsi les aiguiller plus rapidement vers le dispositif de prise en charge. Ces fonctions ont été présentées précédemment, et se répartissent entre le Transputer et le FPGA.

III.1.2.2.1.4. SSCS signalisation

La signalisation s'appuie sur un service AAL, avec une sous-couche SSCS normalisée par l'ITU-T [Q.2100]. Cette sous-couche est divisée en deux parties (cf. figure III.20) :

- SSCOP qui offre un service assuré.
- SSCF (Service Specific Coordination Function) qui offre le service nécessaire au protocole de signalisation.

Le protocole SSCOP [Q.2110] a été présenté dans le paragraphe précédent.

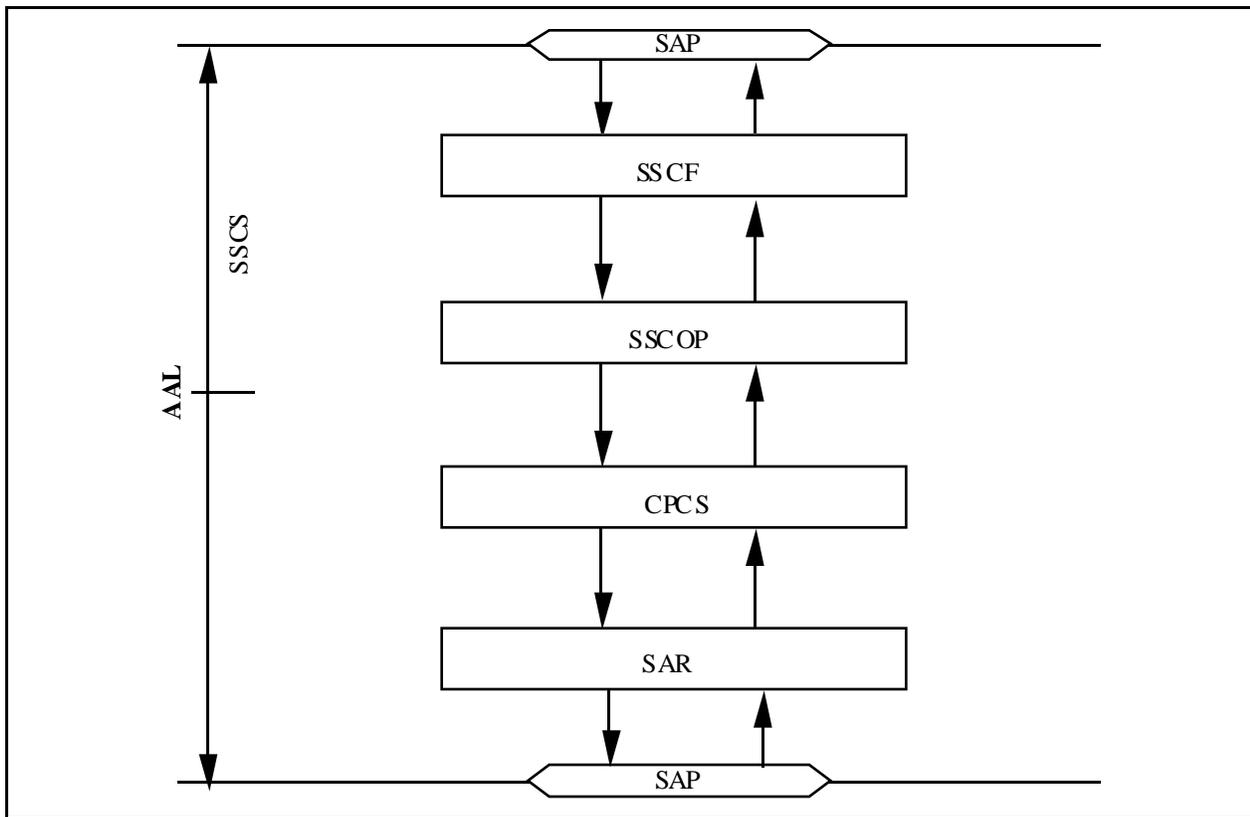


Figure III.20 : La structure détaillée de la couche AAL de signalisation

Plusieurs sous-couches SSCF sont définies en fonction du type du protocole supérieur. Pour la signalisation UNI (User-Network Interface), c'est la norme [Q.2130], pour la signalisation NNI (Network-Network Interface), c'est [Q.2140]. Les fonctions de cette sous-couche sont liées à la gestion des connexions de signalisation, réservées dans ATM. Deux types de connexions sont utilisées : à diffusion ou point à point. Les connexions à diffusion permettent un transfert de l'information non acquitté alors que les connexions point à point offre un transfert assuré.

III.1.2.2.1.5. Le support des autres protocoles

Les protocoles comme IP sur ATM [Col95] ou le LAN Emulation [Cla94] n'ont pas besoin de service SSCS et utilisent directement le service de couche CPCS. Une couche SSCS nulle est donc prévue à cet effet.

III.1.2.2.2. Les protocoles supportés

Ce paragraphe décrit les protocoles supportés au-dessus du service SSCS de la couche AAL. Il s'intéresse aussi aux adaptations nécessaires à apporter aux protocoles standards tels TCP/IP ou OSI.

III.1.2.2.2.1. La signalisation et la méta-signalisation

La signalisation et la méta-signalisation étant des fonctions globales du système de communication, elles sont présentées dans la partie III.2.

III.1.2.2.2.2. IP sur ATM

Pour pouvoir utiliser les applications classiques de téléinformatique (telnet, ftp, client WWW, ...), il est souhaitable d'implanter le protocole TCP/UDP/IP au-dessus d'ATM. Un groupe de travail IETF (Internet Engineering Task Force) étudie l'adaptation d'IP sur ATM [Col95]. La couche AAL utilisée est celle de type 5.

Dans le cadre de ces travaux, deux types de modèles d'architecture de réseau ATM correspondant à deux solutions différentes d'intégration de IP avec un réseau ATM ont été définis :

- Les modèles "subnet models" qui reconnaissent, grâce à la résolution de l'adresse ATM, les systèmes finaux directement connectés, cette adresse étant vue par le réseau IP comme une adresse du niveau liaison de données. L'avantage de ces modèles consiste en leur transparence vis-à-vis de tous les protocoles du niveau réseau et en la réduction de la taille des tables de routage au niveau du réseau "backbone".
- Les modèles "peer models", dans lesquels le réseau ATM agit comme un routeur logique IP en échangeant les informations de routage avec d'autres routeurs IP. L'adresse IP est encapsulée dans la structure NSAP de l'adresse ATM. Ces modèles sont bien adaptés à l'interconnexion de systèmes qui supportent un seul protocole réseau (ex. IP) et requièrent moins d'effort de configuration, d'administration des adresses et de services de support (comme les serveurs ARP par exemple).

Les premiers suivent l'architecture IP traditionnelle alors que les seconds proposent une nouvelle façon de réaliser l'interconnexion IP. Des modèles intermédiaires utilisant des

parties des deux premiers ont été proposés dans [Lia93]. Nous nous intéressons ici seulement au premier type de modèle d'architecture, puisque c'est le plus avancé actuellement. Le but est de supporter les applications de téléinformatique, et non de définir une implantation optimale de IP sur ATM. Des travaux sur les "peer models" sont présentés dans [Hei94] et [Lia93].

Les différents modèles identifiés dans les "subnet models" sont les suivants :

- Le modèle IP classique.
- Le WAN ATM basé sur des SVC.
- Les modèles de sous-réseaux "légers", par exemple TUNIC (TCP and UDP over a Nonexistent IP Connection) et TULIP (TCP and UDP over Lightweigh IP).

Dans le cas du réseau local que nous définissons, c'est le modèle IP classique qui nous intéresse. [Col95] présente les autres modèles.

III.1.2.2.2.1. Le modèle IP classique

Le modèle IP classique [RFC1577] est souvent référencé sous le terme LIS (Logical IP Subnetwork). Dans cette configuration, les stations connectées au même LIS communiquent directement entre elles. Par contre, toute communication avec une station extérieure au LIS s'effectue par un routeur IP. Ce routeur est attaché au réseau ATM et est membre d'un ou plusieurs LIS. Ainsi, des stations de différents sous-réseaux IP doivent communiquer via un routeur IP même s'il est possible d'établir un VC-ATM direct entre les deux membres de la communication IP.

Les équipements (routeurs et stations) utilisant IP sur ATM dans un même LIS ont les caractéristiques décrites ci-après. Tous les membres du LIS :

- Ont la même adresse de sous-réseau IP et le même masque.
- Sont directement connectés au réseau ATM.
- Doivent disposer d'un mécanisme de résolution d'adresses IP en adresses ATM via ATMARP (voir paragraphe III.1.2.2.2.3).
- Doivent supporter les connexions SVC (Switched Vitual Connections) et PVC (Permanent Virtual Connections).

Chaque station supportant IP doit disposer des deux paramètres suivants :

- L'adresse ATM de la station.
- L'adresse du serveur ATMARP du LIS. Lorsque seules les connexions virtuelles permanentes (PVC) sont utilisées, ce paramètre n'est pas nécessaire.

III.1.2.2.2.2. L'encapsulation des datagrammes IP sur AAL 5

Deux méthodes d'encapsulation ont été définies dans [RFC1483]. La première permet le multiplexage de différents protocoles sur un seul chemin virtuel ATM. Le protocole du PDU transféré est identifié en préfixant le PDU par un en-tête LLC IEEE 802.2. Cette méthode est appelée "LLC encapsulation". La seconde méthode utilise un chemin virtuel ATM (VC) par protocole et simplifie donc l'en-tête ajouté. Cette deuxième méthode est nommée "VC based multiplexing".

La méthode "VC based multiplexing" devrait dominer dans des environnements où la création dynamique d'un grand nombre de VC est rapide et peu coûteuse, c'est à dire dans les réseaux privés. Par contre, l'autre méthode peut être utilisée quand il n'est pas pratique d'avoir un VC par protocole. C'est le cas par exemple lorsque le réseau ATM supporte seulement les connexions virtuelles (semi-) permanentes (PVC), ou lorsque le taux de charge du réseau dépend fortement du nombre de connexions actives. Quand deux stations ATM veulent échanger du trafic sans connexion IP, la sélection de la méthode d'encapsulation est faite soit manuellement à la configuration (dans le cas des PVC) ou par la signalisation (dans le cas des SVC).

La taille maximum de l'unité qui peut être transmise pour IP (MTU : Maximum Transmission Unit) est fixée à 9180 octets. Cependant, [RFC1626] propose des tailles supérieures qui pourraient utiliser pleinement les capacités offertes par l'AAL5 (65535 octets). Dans ce cas, la taille de la MTU devra être négociée entre les deux stations participant à la communication. Ceci n'est possible que dans le cas où le réseau supporte les SVC.

III.1.2.2.2.3. La résolution d'adresses

Des mécanismes pour la résolution et la conversion des adresses IP en adresses ATM au format NSAP (défini par l'ATM Forum) ont déjà été proposés. La résolution des adresses entre le niveau réseau et un niveau inférieur (MAC par exemple) se fait normalement en utilisant des protocoles spécifiques (ARP pour Address Resolution Protocol). Les ARP utilisent habituellement des méthodes conventionnelles qui sont basées sur la diffusion des trames ARP. Comme il n'est pas possible d'utiliser la diffusion dans un réseau ATM, la résolution d'adresses dans un réseau IP sur ATM [RFC1577] peut

utiliser le protocole ATMARP (ATM Address Resolution Protocol) (basé sur [Plu82]) et le protocole InATMARP (Inverse Address Resolution Protocol) (basé sur [Bra92]). La résolution d'adresses diffère suivant la disponibilité sur le réseau ATM de connexions virtuelles permanentes (PVC) ou dynamiques (SVC). La gestion des connexions (établissement des SVC) est présentée dans [RFC1755].

En conclusion, pour supporter IP sur ATM, il faut ajouter au protocole IP classique les fonctions de résolution d'adresses adaptées à ATM.

III.1.2.2.2.3. Les autres protocoles

D'autres protocoles pourront s'avérer utiles dans un futur proche, en particulier, certains protocoles du monde OSI. L'ATM Forum définit le concept de LAN Emulation [Cla94] pour supporter ces protocoles. Nous nous contenterons de suivre l'avancement de la normalisation.

Par contre, dans le milieu des applications distribuées temps réel, la messagerie MMS [ISO9506] est très utilisée. Afin de pouvoir supporter les applications définies au-dessus de ce service de la couche 7 du modèle OSI, il est nécessaire d'adapter l'interface offerte. La couche MMS s'appuie essentiellement sur le service [ISO8649] et le protocole [ISO8650] de contrôle d'association ACSE, qui constitue un module de la couche application. Ces services concernent l'établissement, la terminaison et la coupure d'association. Une couche d'adaptation à MMS doit offrir ces services. Elle doit traduire les requêtes sur les associations (point à point et point à multipoint) en requêtes sur les connexions adressées à la couche de signalisation. Elle doit donc gérer une table d'équivalence entre les numéros d'association et les numéros de connexion. Elle est similaire au module nécessaire pour accéder au service de transport rapide. Ceci est valable lorsque les couches session et présentation ne sont pas utilisées, ce qui est généralement le cas. Un complément d'étude est nécessaire pour implanter ces idées.

III.1.2.2.3. Les interfaces offertes aux applications

Ce paragraphe présente les différentes interfaces offertes aux applications (API), en particulier celles directes au-dessus de l'AAL.

III.1.2.2.3.1. L'accès au transport rapide

Pour accéder au protocole de transport rapide directement au-dessus de l'AAL, une application doit pouvoir établir et fermer une connexion avec une application distante,

négozier la qualité de service, et envoyer et recevoir des données. Les primitives applicatives relatives à l'établissement et à la fermeture des connexions point à point que nous avons définies sont :

`open_req (calling_app_id, called_app_id, qos, connection_id)`

Cette primitive permet à une application (identifiée par `calling_app_id`) de demander l'établissement d'une connexion point à point avec une application distante (identifiée par `called_app_id`). Le paramètre `called_app_id` doit être associé à l'adresse ATM de l'appelé. Cette primitive retourne un identificateur de connexion.

`open_ind (calling_app_id, called_app_id, qos, connection_id)`

Cette primitive permet d'indiquer à une application (identifiée par `calling_app_id`) qu'une application distante (identifiée par `called_app_id`) demande l'établissement d'une connexion avec elle.

`open_resp (connection_id, accept)`

Cette primitive permet d'accepter ou de refuser la demande d'établissement de connexion.

`open_conf (connection_id, accept)`

Cette primitive permet d'indiquer à l'application appelante que la demande d'établissement de connexion a été acceptée ou refusée.

`close_req (connection_id)`

Cette primitive permet à une application de demander la fermeture d'une connexion.

`close_ind (connection_id, cause)`

Cette primitive permet d'indiquer à l'application que la connexion a été fermée en raison de la cause spécifiée.

Les primitives relatives à la gestion des connexions point à multipoints sont :

mopen_req (calling_app_id, list_called_app_id, qos, connection_id)

Cette primitive permet à une application (identifiée par calling_app_id) de demander l'établissement d'une connexion point à multipoints avec les applications distantes (identifiées par list_called_app_id). Chaque élément de list_called_app_id doit être associé à l'adresse ATM de l'appelé. Cette primitive retourne un identificateur de connexion.

add_party_req (connection_id, called_app_id)

Cette primitive permet d'ajouter une application (identifiée par called_app_id) à une connexion point à multipoints.

add_party_conf (connection_id, called_app_id, accept)

Cette primitive permet d'indiquer à l'application appelante que l'ajout de l'application (identifiée par called_app_id) à la connexion point à multipoints a été accepté ou refusé.

drop_party_req (connection_id, called_app_id)

Cette primitive permet de retirer une application (identifiée par called_app_id) d'une connexion point à multipoints.

drop_party_conf (connection_id, called_app_id, accept)

Cette primitive permet d'indiquer à l'application appelante que le retrait de l'application (identifiée par called_app_id) de la connexion point à multipoints a été accepté ou refusé.

Les primitives permettant de renégocier la qualité de service après que la connexion ait été ouverte sont :

modify_qos_req (connection_id, qos)

Cette primitive permet à une application de modifier les paramètres de qualité de service d'une connexion. Elle peut servir à modifier le débit de la

connexion, par exemple en cas de changement de qualité de la vidéo. Pour être utilisée, le réseau doit supporter cette fonctionnalité.

`modify_qos_ind (connection_id, qos)`

Cette primitive permet de notifier une application d'une demande de modification de la qualité de service d'une connexion.

`modify_qos_resp (connection_id, accept)`

Cette primitive permet d'accepter ou de refuser la modification de la qualité de service.

`modify_qos_conf (connection_id, accept)`

Cette primitive permet d'indiquer à l'application que la modification de qualité de service a été acceptée ou refusée.

Les deux dernières primitives sont celles concernant directement le transfert des informations :

`send (connection_id, buffer, status)`

Cette primitive permet à une application d'envoyer des données sur une connexion ouverte.

`receive (connection_id, buffer, status)`

Cette primitive permet à une application de recevoir des données sur une connexion ouverte.

III.1.2.2.3.2. L'accès aux transferts critiques

L'interface offerte aux applications pour le transfert périodique de variables est spécifique à notre système. Elle sert uniquement à initialiser et clore le service. En effet, une fois la négociation effectuée, l'application n'intervient plus. Avant d'initialiser un producteur ou un consommateur, une connexion doit être établie entre les deux acteurs, avec les qualités de service requises. Les primitives pour l'ouverture et la fermeture des

connexions entre producteurs et consommateurs sont les mêmes que celles définies dans le paragraphe précédent. Les autres primitives pour l'initialisation du service sont :

Pour le producteur de variables :

```
init_producer (refresh_cycle, connection_id1, ..., connection_idn,  
              (var_name1, mem_address1), ..., (var_namen, mem_addressk))
```

Cette primitive permet au producteur d'initialiser le dispositif qui se charge de transférer les données. Le producteur spécifie le cycle de rafraîchissement pour un ensemble de variables, et un ensemble de points d'accès ATM ouverts sur lesquelles ces variables doivent être transmises. Il donne pour chaque variable son nom et sa localisation.

```
stop_producer (connection_id1, ..., connection_idn)
```

Cette primitive permet au producteur d'arrêter le transfert.

Pour le consommateur de variables :

```
init_consumer (refresh_cycle, connection_id, (var_name1, mem_address1), ..., (var_namen,  
mem_addressn))
```

Cette primitive permet au consommateur d'initialiser le dispositif qui se charge de transférer les données. Le consommateur spécifie le cycle de rafraîchissement qu'il souhaite, et pour chaque variable qu'il veut recevoir sur un point d'accès ATM donné l'adresse mémoire de cette variable.

```
stop_consumer (connection_id)
```

Cette primitive permet au consommateur d'arrêter le transfert.

Les primitives applicatives pour le transfert audio/vidéo sont construites de manière similaire aux précédentes, avec une simplification due au transport d'un seul flux par connexion :

Pour la source audio/vidéo :

```
init_source (connection_id, serv_param, address)
```

stop_source (connection_id)

Pour le puits audio/vidéo :

init_sink (connection_id, serv_param, address)

stop_sink (connection_id)

De même, avant d'initialiser une source ou un puits, une connexion ATM doit être établie entre les deux acteurs, avec les qualités de service requises. Pendant l'initialisation, la source doit définir sur quelle connexion le transfert va s'opérer et quelle est l'adresse de début du tampon des données audio/vidéo. Au niveau du puits, il faut définir l'adresse de début du tampon des données audio/vidéo. Des paramètres de service supplémentaires à ceux définis lors de l'ouverture de la connexion peuvent être négociés.

III.1.2.2.3.3. TCP et UDP

TCP et UDP offre une interface de type "Socket" ou de type "Streams" [Tan90], en fonction de la pile de protocole utilisée. Elle est indépendante de notre système de communication, puisque la pile TCP/UDP/IP sur le système d'exploitation de la machine hôte est utilisée.

III.1.2.3. Résumé

Ce paragraphe a présenté les caractéristiques logicielles de l'architecture de couplage des équipements. Un modèle protocolaire particulier a été défini afin de supporter les nombreux services présents dans le système, que ce soit les services spécifiques de transport rapide et de transferts critiques, la signalisation ou l'utilisation des protocoles existants (TCP/IP) (*caractéristique C5*).

III.1.3. Conclusion

Cette partie a présenté l'**architecture de couplage des équipements** définie pour répondre aux besoins du monde industriel. Les trois aspects importants de cette architecture sont :

- La présence d'un commutateur embarqué sur la carte de communication ATM qui permet de supporter plusieurs équipements audio/vidéo sans passer par le bus fond de panier de la machine, d'offrir des topologies de réseau variées, et de

proposer la redondance de chemins. Le tout permet de construire des réseaux compacts, sans machine de commutation externe.

- La présence d'un micro-contrôleur sur la carte de communication qui permet d'implanter au travers du couple Transputer + FPGA des services temps réel de haut niveau au plus près du réseau afin d'optimiser les transferts, par la notion de télé-DMA.
- Le modèle protocolaire avec les couches SSCS transport rapide et transferts critiques, et le support des protocoles existants (TCP/IP).

Une fois l'architecture de couplage des équipements définie, un certain nombre d'éléments reste à étudier pour construire le système de communication. Le paragraphe suivant s'intéresse à ces problèmes, et plus globalement aux fonctions de gestion de la communication entre les équipements, avec notamment le contrôle de congestion, la signalisation et la gestion de la redondance.

III.2. La gestion de la communication entre les équipements

Nous avons vu jusqu'alors l'architecture de couplage des équipements. Pour que le système puisse fonctionner correctement, il manque quelques éléments. Ce chapitre s'intéresse à la gestion de la communication entre les équipements. Les fonctions sont de plusieurs ordres, et font chacune l'objet d'un paragraphe :

- Le contrôle de congestion sur ATM est indispensable pour offrir un acheminement correct des informations et garantir la qualité de service, en particulier borner le délai de transmission.
- Le support des services en mode non connecté au-dessus du réseau ATM est nécessaire. Il faut veiller tout particulièrement à ce que celui-ci ne détériore pas le trafic en mode connecté.
- La signalisation est indispensable pour négocier, ouvrir et fermer les connexions ATM. Elle est particulière dans notre cas à cause du commutateur embarqué et de la présence de plusieurs cartes par station.
- L'administration du réseau permet de suivre le fonctionnement du réseau et de gérer sa structure.
- La gestion de la redondance au niveau du réseau et la reconfiguration satisfont une partie des problèmes de sûreté.

III.2.1. Le contrôle de congestion

III.2.1.1. Introduction

Les réseaux ATM disposent de ressources limitées pour assurer l'acheminement des cellules. Ces limitations se situent au niveau de la capacité des lignes physiques et de la capacité de traitement des commutateurs (taille des tampons ou des files d'attente). Quand la demande de trafic croît, le réseau subit des phénomènes de congestion similaires à ceux observés en circulation automobile. Ces phénomènes, s'ils ne sont pas traités, mènent à un effondrement des performances du réseau et au non respect des qualités de service. Mais, à la différence de la circulation automobile, des informations peuvent être perdues.

Lorsque nous avons mené cette étude, de nombreuses recherches sur le contrôle de congestion étaient en cours, mais aucune solution répondant à toutes les formes de trafic n'avait été trouvée. Il n'en existe d'ailleurs toujours pas actuellement (un descriptif des méthodes proposées peut-être consulté dans l'annexe B). Par conséquent, nous avons cherché une solution correspondant à notre problème de transfert d'informations temps

réel, où il faut éviter toute perte et garantir un délai d'acheminement borné des messages. Ainsi, il faut éviter toute congestion, car une congestion au niveau d'un commutateur se traduit par la perte de cellules. La perte d'une cellule signifie la perte de tout un message. Celui-ci devra être réémis si la connexion est en mode assuré, ce qui implique un surcroît de travail non négligeable. De plus, il sera impossible de borner le délai d'acheminement d'un message, ce qui est indispensable pour les applications temps réel.

Les problèmes d'accès au réseau et de congestion ont des solutions bien connues dans le cas d'un anneau avec des utilisations de la bande passante plus ou moins efficace (Token Ring [802.5], FDDI [Ros86], anneau à compensation [Sab93]). Par contre, il est beaucoup plus difficile de trouver une solution valable pour tout type de topologie, qu'elle soit en bus, en anneau, complètement maillée ou surtout quelconque.

Ce paragraphe s'intéresse à la recherche d'une **solution pour une topologie quelconque**. En premier lieu, il présente le type de contrôle d'accès au réseau choisi. Puis, il décrit une méthode pour éviter les congestions. Finalement, il montre comment faire cohabiter les différents trafics (temps réel, données traditionnelles, ...) sur le réseau.

III.2.1.2. Le contrôle d'accès au réseau

Plusieurs politiques de contrôle d'accès sont possibles. Un contrôle d'accès strict a pour conséquence de rejeter toutes les cellules correspondant au trafic en excès. Des contrôles d'accès plus souples ont été proposés, par exemple en acceptant quelques violations de contrat (Leaky Bucket [Tur86, Lem93]), ou en utilisant le bit CLP et en marquant les cellules en excès. Ces dernières seront éliminées seulement lors d'une congestion.

Dans l'architecture définie dans la partie III.1, nous disposons d'un composant AAL effectuant un contrôle de flux selon la méthode du Leaky Bucket. Deux contrôles sont effectués, l'un par connexion et l'autre globalement sur l'ensemble des connexions. Ceci permet d'espacer le flux émis sur le réseau. Par contre, aucun autre dispositif de régulation du flux n'est présent sur le réseau. Il est donc nécessaire d'effectuer un **contrôle de flux strict à la source**. Dans ce cas, les cellules seront envoyées sur le réseau uniquement si elles respectent le débit et la taille des rafales. Les commutateurs disposent cependant de files d'attente en sortie d'une capacité de 75 cellules chacune, qui peuvent permettre d'absorber de petites variations du trafic.

Sous ces hypothèses, nous allons chercher quelles sont les conditions qui peuvent garantir qu'il n'y aura pas de congestion. Pour cela, il faut s'assurer qu'aucune rafale ne

puisse saturer une file d'attente d'un commutateur. Le paragraphe suivant présente une solution.

III.2.1.3. L'évitement de congestion

Le flux global de chaque source est contrôlé par un espaceur. Celui ci génère un flux dont la taille des rafales est bornée. Le problème de simultanéité de trafic se réduit ainsi au nombre de sources et non plus au nombre de connexions. A une exception près : il faut considérer les chemins physiques qu'utilise chaque source. Une source peut en effet utiliser plusieurs chemins physiques différents pour acheminer les cellules de ses connexions. Elle peut aussi utiliser le même chemin pour plusieurs connexions.

Nous nous plaçons sur une topologie quelconque construite à partir des commutateurs 4x4 embarqués sur les cartes présentées la partie III.1. Nous supposons que les connexions allant sur une sortie d'un commutateur peuvent provenir des quatre entrées de ce même commutateur. Ceci n'est valable que pour deux des sorties. En effet, la sortie allant vers le composant AAL n'aura jamais de trafic en provenance de l'entrée associée (une communication locale ne passe pas par le commutateur ATM). Il en est de même pour la sortie vers le bus local ATM. Cependant, nous garderons l'hypothèse avec quatre entrées puisqu'elle est plus générale, et donc plus restrictive.

Soient :

- b_1, b_2, b_3, b_4 les rafales arrivant simultanément sur les entrées e_1, e_2, e_3, e_4 pour la sortie s (cf. figure III.21).
- TB la capacité totale de la file d'attente de la sortie s .
- t le temps de réception des quatre rafales. t correspond au temps de réception de la plus longue rafale.

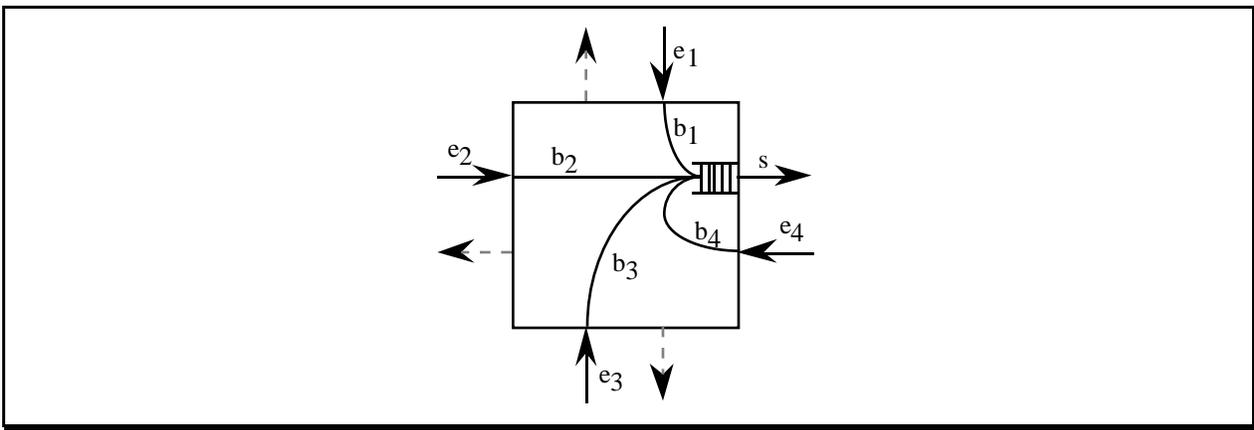


Figure III.21 : Les entrées/sorties d'un commutateur

Pendant le temps t , la sortie s a le temps d'expédier le même nombre de cellules que celui de la plus longue rafale. Pour assurer qu'il n'y ait aucune perte, il faut donc pouvoir stocker le nombre de cellules correspondant à la somme des trois plus petites rafales.

Ceci se traduit par l'équation suivante :

$$b_1 + b_2 + b_3 \leq TB \quad \text{avec } b_1 \leq b_4 \text{ et } b_2 \leq b_4 \text{ et } b_3 \leq b_4 \quad (1)$$

La période d'observation étant égale à t , nous avons donc :

$$b_4 \leq t \quad (2)$$

Ainsi, d'après (1) + (2) :

$$b_1 + b_2 + b_3 + b_4 \leq 4t \quad (3)$$

Pour accepter des rafales de longueur maximale sur chaque entrée, il faut pouvoir avoir :

$$b_1 = t \text{ et/ou } b_2 = t \text{ et/ou } b_3 = t \quad (4)$$

D'où, d'après (1) + (4) :

$$3t \leq TB \quad (5)$$

Nous avons donc, d'après (3) + (5) :

$$b_1 + b_2 + b_3 + b_4 \leq \frac{4}{3} TB \quad (6)$$

En d'autres termes, la somme des longueurs des rafales arrivant sur les quatre entrées doit être au plus égale aux $\frac{4}{3}$ de la taille d'une file d'attente.

Il faut maintenant caractériser la taille maximale d'une rafale en fonction du nombre de chemins. Les rafales peuvent être émises par une source, mais elles se créent principalement par la simultanéité d'émission des sources. Comme il y a des retards possibles entre différents chemins, une source peut envoyer des cellules sur des chemins différents qui pourront former une rafale sur un chemin commun. Donc la taille maximale d'une rafale sur un lien est le nombre de chemins différents qui empruntent ce lien, dans le cas où les sources n'émettent pas de rafale.

Plaçons nous dans le cas où l'espaceur peut éviter de coller les cellules, c'est-à-dire dans le cas où le trafic émis par une station ne dépasse pas la moitié du débit du réseau (soit environ 78 Mbit/s). Nous verrons comment supprimer cette contrainte par la suite. Sur une entrée d'un commutateur, on peut observer une rafale de n cellules si le trafic sur ce lien provient de n sources. Avec le commutateur utilisé, disposant de files de 75 cellules, il est possible de faire passer 100 chemins différents par une même sortie d'un commutateur.

Le nombre de connexions passant par une sortie d'un commutateur peut donc être supérieur ou égal à 100. Ceci pourrait être gênant si toutes les stations devaient communiquer en permanence avec toutes celles présentes sur le réseau. Sous cette condition et avec une topologie en anneau simple, le nombre de stations serait alors limité à 100. Dans tous les autres cas, ce nombre serait supérieur.

Cependant, si cette politique de contrôle de congestion ne limite pas trop le nombre de stations ou de connexions, elle restreint la liberté pour l'envoi des cellules au niveau des sources. En effet, nous nous sommes placés dans le cas où chaque source n'émet pas de rafale. Pour faire sauter cette contrainte, il faut se rappeler que c'est la taille maximale des rafales allant vers une même sortie qui entre dans les calculs. Pour permettre à une station d'émettre des rafales de longueur n , il suffit de diminuer le nombre de chemins possibles de la valeur n .

Pour gérer cette politique de congestion, il est nécessaire de disposer d'informations au niveau des commutateurs. Chaque commutateur doit avoir, pour chaque sortie, les informations suivantes :

- Le débit utilisé par rapport au débit total du lien.
- Le nombre de chemins passant par cette sortie, ainsi que la description de chaque chemin sous la forme d'une suite de couple (numéro du commutateur, port de sortie).
- La longueur totale des rafales passant par cette sortie. Cette valeur est calculée en ajoutant la longueur de la rafale autorisée sur chaque chemin.

Ces informations sont mises à jour lors de la négociation des connexions ATM (ouverture, fermeture, changement de paramètres en cours de communication).

Cette méthode de contrôle de congestion n'est pas très flexible mais elle permet d'éviter tout débordement de files. Le paragraphe suivant montre comment ajouter un peu de flexibilité au système, par exemple pour le transfert de données sans contrainte de délai (le transfert de fichier par exemple).

III.2.1.4. La gestion des trafics

Les flux temps réel ont besoin de connaître le délai maximum d'acheminement d'un message. La politique de gestion des congestions présentée ci-avant permet d'obtenir cette borne. Au niveau du réseau (au niveau de l'AAL), le retard pour un message est défini par la formule suivante :

$$\text{Borne} = N_{\text{com}} \times (\text{TB} \times T_{\text{cell}} + T_{e/s})$$

avec TB : taille d'une file d'attente.

N_{com} : nombre de commutateurs traversés.

T_{cell} : temps de traitement d'une cellule dans un commutateur.

$T_{e/s}$: temps d'entrée et de sortie sur une carte ATM (hors traitement dans le commutateur).

Pour un nombre de commutateurs traversés égal à 10, la borne du retard est égal à 2,1 millisecondes environ. Cette borne est calculée dans le cas où toutes les ressources du réseau sont utilisées, c'est à dire qu'il n'y a plus de chemin possible sans risque de débordement de file. Cette borne est atteinte quand la dernière cellule d'un message doit sortir de chaque commutateur par une file qui est pleine. Ce cas est très improbable.

Par contre, les flux sans contrainte de délai comme le transfert de fichiers pourraient profiter d'une politique plus souple. En effet, les flux temps réel ne remplissent pas

l'ensemble de la bande passante. Il faut néanmoins assurer que les flux sans contrainte de délai ne gênent pas les autres trafics. En particulier, il ne faut pas qu'ils impliquent la perte de données temps réel. Pour cela, le bit CLP de la cellule ATM peut être utilisé. Il permet d'indiquer une priorité à la perte. Les flux sans contrainte de délai ont alors le bit de priorité à la perte à 1. Ainsi, si une cellule d'un flux temps réel doit être placée dans une file d'attente pleine, une cellule avec le bit CLP à 1 est remplacée par la cellule entrante. Ceci permet de mieux utiliser la bande passante. En contre partie, il n'y a pas de garantie d'acheminement d'un message complet pour cette classe de flux. Seules des simulations peuvent montrer le nombre de pertes de cellules en fonction de la charge du réseau.

III.2.1.5. Résumé

En résumé, ce paragraphe a défini une politique de contrôle de congestion qui permet d'éviter toute perte de cellules pour une certaine classe de trafic. Il est cependant possible de faire circuler du trafic non garanti sans perturber la première classe, afin de donner plus de souplesse à l'acheminement de données, et de mieux utiliser la bande passante. Dans l'état actuel de la normalisation sur le contrôle de congestion, il n'existe pas de méthode permettant de garantir un acheminement dans un délai borné d'un message. Si une telle solution apparaissait, il serait possible de l'utiliser.

Le paragraphe suivant s'intéresse au service non connecté au-dessus d'ATM.

III.2.2. Le mode non connecté

III.2.2.1. Introduction

Le réseau public ATM est constitué d'un ensemble de commutateurs interconnectés par des liens série à haut débit. Le transport des informations se fait en mode connecté. Pour supporter le mode non connecté, un réseau de connexions virtuelles de machines serveurs de données sans connexion est construit au-dessus du réseau ATM. Il fonctionne avec des protocoles particuliers (SMDS [Bel91] aux États Unis et CBDS [Ets92] en Europe).

Un réseau local, à l'opposé, doit avoir une structure entièrement répartie et distribuée. Les commutateurs doivent combiner un faible coût, une taille réduite, une puissance et une maintenabilité importante [Bry92]. La gestion des connexions doit être entièrement répartie sur tous les nœuds du réseau. D'autre part, le mode non connecté ne peut pas être géré de manière similaire au réseau public. En effet, un serveur sans connexion représente une quantité de trafic supplémentaire trop importante pour en

justifier l'utilisation. Il faut donc définir d'autres mécanismes pour la gestion des données sans connexion.

A cette fin, la communauté Internet (IETF : Internet Engineering Task Force) a défini IP sur ATM [Col95] de manière à supporter le protocole sans connexion IP sur l'AAL 5. D'un autre côté, l'ATM Forum a élaboré le concept de LAN Emulation [Lem95] pour récupérer certains protocoles de niveau MAC.

Les deux paragraphes suivants présentent les grandes lignes de ces concepts, et discutent leurs avantages et leurs inconvénients.

III.2.2.2. IP sur ATM

Plusieurs modèles ont été proposés pour l'adaptation du protocole IP sur ATM (cf. paragraphe III.1.2.2.2.3.). Le modèle IP classique nécessite trois éléments :

- Un client.
- Un serveur de résolution d'adresses (serveur ATMARP).
- Un (ou plusieurs) routeurs pour interconnecter les différentes zones.

Dans ce modèle, un client voulant envoyer un message doit s'adresser au serveur de résolution d'adresses pour convertir l'adresse IP en adresse ATM. Le serveur renvoie l'adresse ATM au client. En fonction de la localisation du destinataire, une connexion ATM sera ouverte directement avec le destinataire si celui-ci est dans le même LIS (Logical IP Subnet) (zone locale) ou avec un routeur s'il appartient à un autre LIS. Une connexion établie n'est pas fermée immédiatement après la fin de l'envoi des données. Celle-ci peut être utilisée ultérieurement. Si ce modèle est proche du modèle actuel sur Ethernet, les communications entre deux clients appartenant à des LIS différents nécessitent de traverser un routeur même si une connexion ATM pouvait être établie directement entre les deux interlocuteurs. Ceci est clairement inefficace, puisque les routeurs ATM deviennent un goulot d'étranglement. De plus, si une qualité de service a besoin d'être négociée, il faut ouvrir une connexion par message, ce qui devient du mode connecté.

Des extensions de ce modèle ont été proposées à l'IETF pour supprimer ces limitations. Le protocole NHRP (Next Hop Resolution Protocol) [Kat94] est en cours de discussion. La notion de réseau NBMA (Non-Broadcast Multi-Access) se substitue au concept de LIS. Un réseau NBMA ne supporte pas de mécanisme de diffusion. C'est le cas pour ATM, Frame-Relay ou X.25. Un tel réseau est constitué d'un ensemble de noeuds, chacun d'eux étant attaché au même réseau NBMA, même s'ils ne communiquent pas

directement. Le serveur ATMARP est remplacé par un serveur NHRP (NHS) qui maintient des tables de conversion d'adresses IP et ATM pour tous les noeuds associés au NHS, et pour les préfixes IP nécessitant le passage par un routeur.

Les détails pour l'encapsulation et la résolution d'adresses ont été présentés dans le paragraphe III.1.2.2.2.3.

III.2.2.3. LAN Emulation

Le second concept pour supporter le mode non connecté sur ATM est le LAN Emulation. Ce protocole définit les mécanismes pour émuler soit Ethernet, soit Token Ring. Il spécifie une interface de niveau réseau (couche 2) identique à celle des réseaux locaux existants et une méthode d'encapsulation dans un format MAC approprié à ATM. Ceci permet d'utiliser les protocoles actuels sans modification, puisque l'interface pour les pilotes de couche réseau est la même (NDIS ou ODI par exemple). Quatre éléments sont nécessaires pour le LAN Emulation :

- Un client (LEC : LAN Emulation Client), pouvant être une interface au réseau ou un commutateur.
- Un serveur de résolution d'adresses (LES : LAN Emulation Server), unique pour un réseau local donné.
- Un serveur de diffusion (BUS : Broadcast and Unknown Server).
- Un serveur de configuration (LECS : LAN Emulation Configuration Server).

Lors de la phase d'initialisation, le client doit obtenir sa propre adresse ATM. Il établit une connexion avec le serveur de configuration (LECS), dont il a précédemment obtenu l'adresse. Le serveur de configuration lui renvoie alors les paramètres nécessaires à son fonctionnement (type de réseau local émulé, taille maximum des PDU, ...). Le client établit ensuite une connexion de contrôle avec le serveur (LES). Cette connexion lui permet d'utiliser le protocole de résolution d'adresses MAC en adresses ATM (LE_ARP). Le client utilise le protocole LE_ARP pour obtenir l'adresse du serveur de diffusion (BUS), avec lequel il établit aussi une connexion.

Pour l'envoi de données, le client utilise aussi le protocole LE_ARP pour obtenir l'adresse ATM du destinataire. Mais la réponse pouvant prendre du temps, les données sont envoyées au serveur de diffusion. Ce dernier les diffuse sur le réseau. Lorsque le client obtient l'adresse ATM, il établit une connexion directe avec le destinataire. La suite des données passe alors par cette connexion, avec au préalable un message de purge vers le serveur de diffusion pour assurer le bon séquençement des données.

Le LAN Emulation est prévu pour utiliser uniquement les services ABR et UBR, puisque ce sont eux qui correspondent le mieux à la nature non-connectée des protocoles MAC. Ceci empêche par exemple aux protocoles supérieurs d'utiliser les services VBR.

III.2.2.4. Résumé

Ce paragraphe a présenté deux méthodes qui ont été définies, l'une par l'IETF, l'autre par l'ATM Forum, pour supporter un service sans connexion sur ATM. Mais ce service ne permet pas de tirer profit des caractéristiques d'ATM, contrairement aux protocoles SMDS ou CBDS.

En effet, le modèle IP classique nécessite l'utilisation de routeurs pour communiquer entre deux sous-réseaux locaux, ce qui annihile les avantages apportés par ATM. De plus, lorsque l'on désire spécifier la qualité de service, il faut alors établir une connexion par message, ce qui revient à un mode connecté.

Le LAN Emulation diffuse les données durant la phase de résolution d'adresse. Ceci entraîne un surcroît de trafic sur le réseau non négligeable, et il convient de traiter rapidement le protocole de résolution de l'adresse. De plus, la notion de qualité de service n'est pas prise en considération.

Néanmoins, ces deux méthodes offrent une solution pour récupérer les applications existantes. Dans notre système de communication, il est possible d'intégrer ces deux méthodes sans dégrader le reste du système. En effet, le contrôle de congestion s'effectuant au niveau des connexions ATM, il n'y a pas d'incidence sur les autres communications. Le LAN Emulation s'appuyant sur un service de type "best effort", c'est bien celui-ci qui est fourni avec la politique d'assouplissement du contrôle du trafic.

III.2.3. La signalisation

III.2.3.1. Introduction

La configuration des connexions applicatives repose sur l'utilisation des services offerts par la signalisation, élément du modèle de référence ATM chargé de la gestion des connexions (établissement, fermeture, respect des qualité de services, ...). La signalisation ATM, dans le cas d'un réseau privé, comporte deux modèles :

- Le modèle UNI (User Network Interface) concerne les messages de signalisation échangés entre l'entité de signalisation d'une station et celle d'un commutateur du réseau. Le modèle UNI est aujourd'hui défini par l'ATM Forum [UNI3.1]. Il ne l'était pas encore lors de notre étude.
- Le modèle NNI (Network Network Interface) concerne les messages de signalisation échangés entre les entités de signalisation des commutateurs du réseau. Une première version rudimentaire de ce protocole a été récemment définie par l'ATM Forum [IISP95]. Mais ce modèle reste pour l'instant lié aux constructeurs de commutateurs ATM, chacun ayant son modèle propriétaire de telle sorte que l'interopérabilité des commutateurs passe actuellement par des connexions virtuelles permanentes ouvertes sur chaque commutateur indépendamment des autres, c'est-à-dire sans échange de message de signalisation.

La figure III.22 situe ces différents modèles dans un réseau ATM.

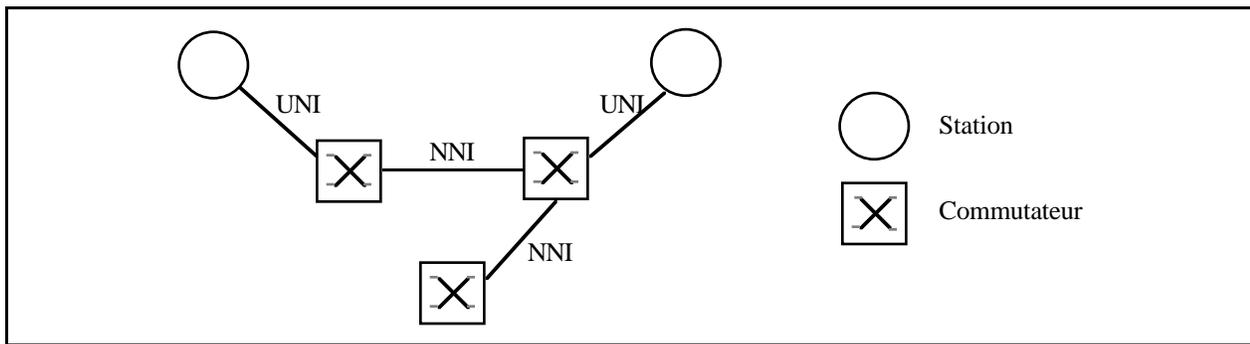


Figure III.22 : Les modèles UNI et NNI

L'architecture originale définie dans la partie III.1 impose une signalisation qui puisse être à la fois UNI et NNI. Ce modèle est décrit dans les sections suivantes.

III.2.3.2. La présentation du modèle

Une des originalités de l'architecture est d'intégrer un commutateur sur la carte d'interface. De ce fait, la signalisation d'une station doit supporter le modèle NNI. De plus, cette architecture doit pouvoir s'interfacer avec une carte ATM classique, c'est-à-dire une carte point à point dont la signalisation respecte le modèle UNI.

L'entité de signalisation devra donc supporter plusieurs connexions de signalisation, certaines avec des entités de signalisation du modèle UNI et d'autres avec des entités de signalisation du modèle NNI. Ce concept est présenté dans la figure III.23.

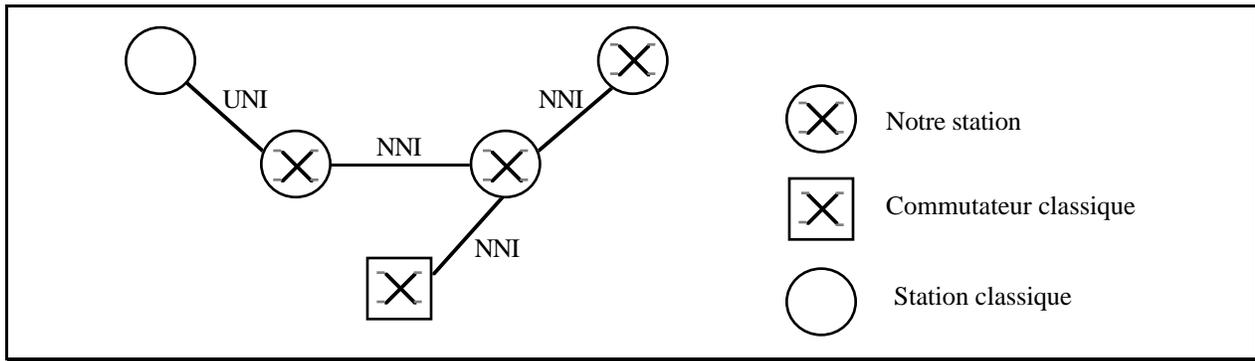


Figure III.23 : Le modèle UNI-NNI

La particularité est donc d'avoir des stations multipoints connectées à la fois à des commutateurs NNI, à des stations multipoints NNI ainsi qu'à des stations point à point classiques UNI. Le modèle est similaire à celui d'un commutateur ATM classique, qui doit s'interfacer à ces 2 types d'équipements. La signalisation répondant à ce concept est décrite dans la section suivante.

III.2.3.3. La signalisation

La signalisation définie ici est basée sur une première version de [Q.93b], aujourd'hui renommé [Q.2931]. Lors de notre étude, sa normalisation au sein de l'UIT-T était à ces débuts. Elle a été étendue afin de traiter les connexions point multipoints.

III.2.3.3.1. Les messages de signalisation

Les messages utilisés constituent un sur ensemble du modèle UNI défini dans [Q.93b] afin d'être compatible avec la norme. Les principaux messages sont les suivants :

- Setup, demande d'ouverture de connexion.
- Connect, réponse positive à une demande d'ouverture de connexion.
- Disconnect, réponse négative à une demande d'ouverture de connexion et fermeture de connexion déjà ouverte.
- Connect Acknowledge, confirmation de l'ouverture de connexion.
- Msetup, demande d'ouverture de connexion point multipoints.

Les principaux paramètres de ces messages, dont une liste plus complète peut être trouvée dans [Q.93b], sont les suivants :

- Numéro de référence pour l'appelant et les appelés. Il identifie de façon unique la connexion.
- Adresse de l'appelant.
- Adresse de l'appelé ou liste des adresses des appelés.
- Paramètres de qualité de services (débit maximum, débit moyen, longueur maximum des rafales, délai d'établissement de la connexion, délai d'acheminement maximum souhaité, ...)

III.2.3.3.2. Le protocole de signalisation UNI

Les messages de signalisation sont échangés sur des connexions de signalisation ouvertes lors de la phase de configuration du réseau entre chaque station voisine. Les connexions de signalisation UNI sont celles reliant une entité de signalisation (SIG) d'une station multipoints ou un commutateur à une SIG d'une station classique.

L'ouverture d'une connexion point à point nécessite les étapes suivantes :

- Demande d'établissement de connexion : une demande d'ouverture de connexion (Setup) est envoyée par la SIG de l'appelant vers la SIG de l'appelé via une SIG voisine avec laquelle elle a une connexion de signalisation.
- Réponse : l'appelé répond par un Connect s'il accepte l'ouverture ou par un Disconnect dans le cas contraire.
- Confirmation : sur réception d'un Connect, la SIG de l'appelant doit envoyer un Connect Acknowledge pour que les pré-réservations effectuées sur le chemin puissent être validées.

L'appelant peut recevoir un Disconnect correspondant à une impossibilité du réseau (signalisation NNI) d'établir la connexion demandée.

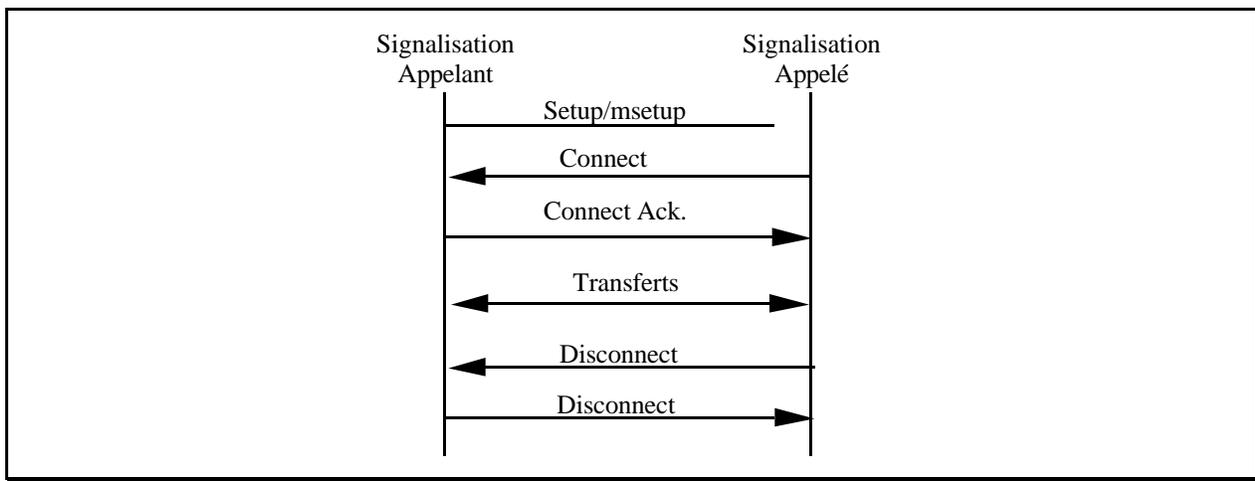


Figure III.24 : Les échanges de la signalisation UNI

L'ouverture d'une connexion point multipoints se différencie côté UNI par l'envoi d'un Msetup comportant la liste des appelés. L'utilisation du Connect et du Connect Acknowledge côté UNI reste inchangée.

III.2.3.3.3. Le protocole de signalisation NNI

Nous avons été amenés à définir notre propre protocole NNI puisqu'aucune normalisation n'existait à ce propos lors de notre étude. Depuis, une première version simplifiée de norme est apparue et il faudra s'adapter à son évolution. Nous présentons ici les principes de la signalisation NNI que nous avons définie.

Les messages de signalisation sont échangés sur des connexions de signalisation ouvertes lors de la phase de configuration du réseau entre chaque station voisine. Les connexions de signalisation NNI sont celles reliant une SIG d'une station multipoints ou d'un commutateur à une SIG d'une autre station multipoints ou d'un autre commutateur.

Le protocole de signalisation NNI se distingue du protocole de signalisation UNI par :

- La nécessité de maintenir à jour des informations concernant la présence des stations connectées sur le réseau.
- La nécessité de choisir entre plusieurs chemins possibles vers un même destinataire.
- La nécessité d'équilibrer la charge du réseau.

Les informations concernant les stations présentes sur le réseau sont maintenues à jour par les agents, d'une part à l'aide des informations fournies par l'administrateur lors

de l'initialisation de toute nouvelle station, et d'autre part à l'aide des flux OAM qui testent le réseau [I.610].

L'établissement d'un chemin est distribué, c'est-à-dire qu'une SIG NNI n'a pas connaissance des ressources des autres nœuds du réseau. Elle connaît simplement les débits disponibles localement, c'est-à-dire les débits disponibles sur chaque sortie menant à une SIG voisine. La sortie choisie est celle ayant le plus grand débit disponible. Après propagation du Setup/Msetup, le débit disponible sur la sortie choisie est décrémenté de la valeur requise par la demande d'ouverture de connexion.

Les messages Setup, Msetup, Connect, Disconnect et Connect Acknowledge du modèle UNI sont utilisés.

Sur réception d'un Disconnect émis par une SIG intermédiaire, une SIG NNI cherche une autre sortie non encore explorée et :

- Envoie un nouveau Setup/Msetup vers l'appelé s'il existe une sortie.
- Envoie un Disconnect vers l'appelant dans le cas contraire.

Le problème de la diffusion de groupe est de pouvoir, en un point du réseau, aiguiller un flot de données vers des destinations différentes (en empruntant des chemins physiques différents). La solution qui consisterait à ouvrir plusieurs connexions point à point est coûteuse et la "multiplication" du flot de données à l'émission n'est pas si simple à réaliser car elle pose des problèmes de temps réel et de synchronisation. Le flot de données doit donc être dupliqué au niveau ATM sans intervention du logiciel, c'est-à-dire au niveau des commutateurs.

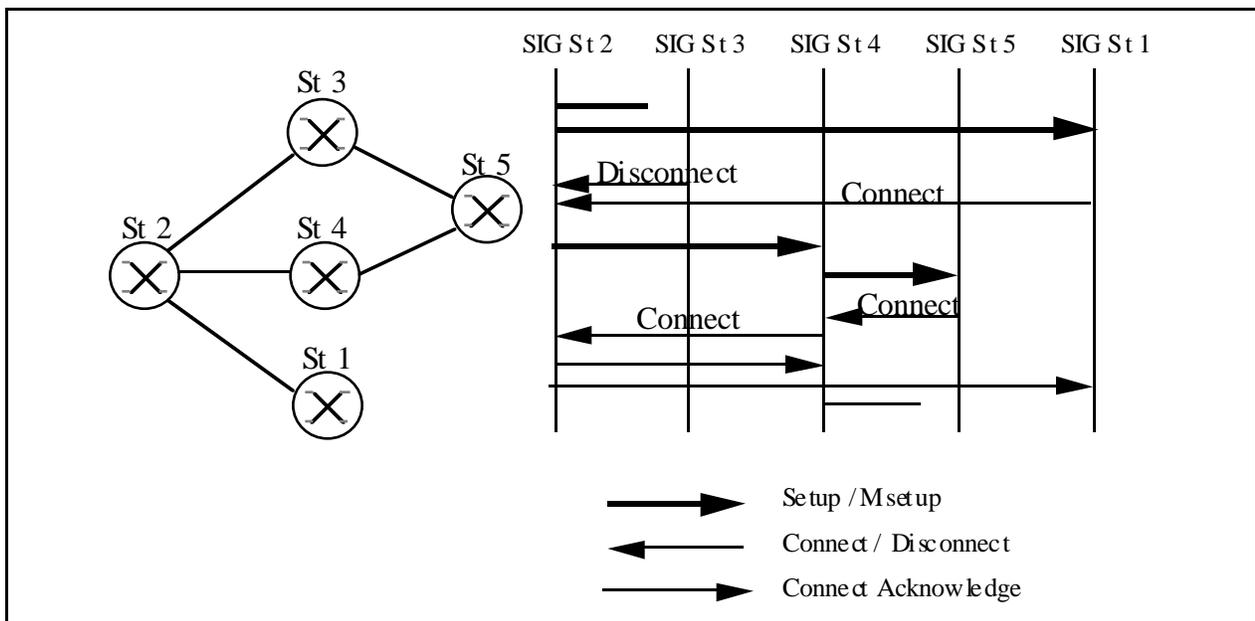


Figure III.25 : Les échanges de la signalisation NNI

La figure III.25 présente les messages échangés dans le cas de l'établissement d'une connexion point multipoints. Sur cet exemple, un Msetup arrive à la SIG de la station 2 (notée St 2) pour une demande d'ouverture de connexion point multipoints avec St 5 et St 1 avec un débit souhaité de 20 Mbit/s. Les débits disponibles sur les liens de St 2 à St 3 et St 2 à St 4 sont respectivement de 100 Mbit/s et de 50 Mbit/s, le débit disponible de St 3 à St 5 est de 10 Mbit/s. Les échanges sont les suivants :

- SIG2 envoie un Msetup à SIG3 d'ouverture de connexion point multipoints avec St 5 et un Msetup à SIG1 d'ouverture de connexion point multipoints avec St 1.
- SIG3 envoie un Disconnect à SIG2 car le débit dont elle dispose vers St 5 est insuffisant.
- SIG1 envoie un Connect à SIG2 (on suppose que l'application appelée accepte l'ouverture).
- SIG2 envoie un Msetup à SIG4 d'ouverture de connexion point multipoints avec St 5.
- SIG4 envoie un Msetup à SIG5 d'ouverture de connexion point multipoints avec St 5.
- SIG5 envoie un Connect à SIG4 (on suppose que l'application appelée accepte l'ouverture).
- SIG4 envoie un Connect à SIG2.
- SIG2 ayant reçu tous les Connect attendus, elle envoie un Connect vers la SIG par laquelle est arrivée la demande (cette SIG et ce message n'apparaissent pas dans la figure III.25).

- SIG2, sur réception du Connect Acknowledge (n'apparaît pas dans la figure III.25), propage ce Connect Acknowledge vers SIG4 et SIG1.
- SIG4 propage le Connect Acknowledge vers SIG5.

Le passage d'un Setup/Msetup dans une station provoque la pré-réservation des débits ainsi que la création d'une ou plusieurs entrées dans la table de routage.

Le passage d'un Connect Acknowledge dans une station provoque la validation des débits réservés. Dès lors, tout message sur la connexion ainsi créée sera dupliqué par le commutateur de St 2 sans passer par les couches logicielles et sera routé automatiquement vers St 4 et St 1.

III.2.3.4. L'ouverture de connexions par une tierce partie

Un protocole important n'existant pas dans la norme actuelle est l'ouverture de connexions par une tierce partie. En effet, une application peut vouloir établir une connexion entre plusieurs équipements qui ne se trouvent pas sur la station sur laquelle elle s'exécute. Par exemple, lorsqu'un opérateur veut, à partir de son poste de configuration, établir une communication entre une caméra et un écran de surveillance. Dans ce cas, l'application fait sa demande à l'agent qui gère la station où elle s'exécute. L'agent effectue alors la demande d'ouverture auprès de la SIG de l'équipement émetteur. Cette SIG se comportera alors comme si elle avait reçu cette demande de l'équipement local, pouvant ouvrir une connexion point à point ou multipoints (cf. figure III.26).

De même, la demande de fermeture de connexion se fait en envoyant le message de fermeture à la SIG de l'émetteur.

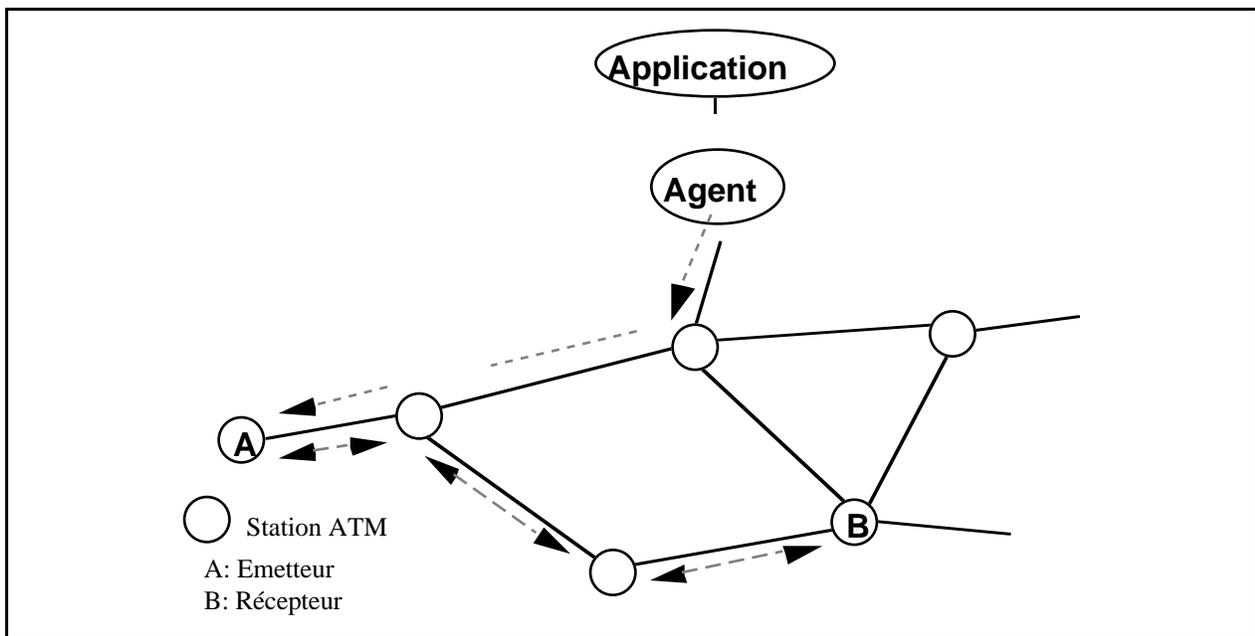


Figure III.26 : L'ouverture de connexions par une tierce partie

III.2.3.5. Résumé

La présence du commutateur embarqué sur notre machine de communication implique la définition d'une signalisation particulière. En effet, notre station doit établir des connexions aussi bien avec des stations classiques au moyen de la signalisation UNI qu'avec des commutateurs ou avec nos stations similaires en utilisant la signalisation NNI.

Outre l'aspect connexion point à point et multipoints, un protocole pour l'ouverture de connexions par une tierce partie a été présenté. Celui-ci est très important dans le domaine d'application visé, où les équipements sont très souvent distants et configurés par un (ou plusieurs) poste de commande.

III.2.4. L'administration du réseau

III.2.4.1. Introduction

L'administration du réseau doit fournir les mécanismes pour configurer, visualiser, contrôler, modifier les ressources à l'intérieur du réseau et doit fournir les services et protocoles standards pour communiquer les informations d'administration. Pour décrire les opérations d'administration, les ressources sont vues comme des objets administrés dont les propriétés et les actions autorisées sont définies. Trois éléments fondamentaux caractérisent une administration de réseau :

- Des fonctions d'administration.
- Des objets administrés.
- Des services et protocoles standards d'administration.

D'autre part, les fonctions d'administration d'un réseau local se divisent en trois parties :

- La **configuration** recouvre l'installation d'un nouveau service, la mise en service d'un nouvel équipement ou des modifications de routage.
- Le **calcul des performances** permet d'évaluer la qualité d'un service, le taux de défaillance d'un équipement ou sa disponibilité.
- La **détection de fautes** traite par exemple de l'impossibilité d'accéder à un service ou d'une alarme émise par un équipement.

III.2.4.2. Le modèle d'administration

Dans un environnement distribué, les entités composant l'administration (appelées applications d'administration) peuvent être elles-mêmes réparties sur l'ensemble du réseau. Elles s'associent simplement pour gérer l'ensemble des informations d'administration. Entre les applications d'administration, circulent des opérations d'administration et des notifications grâce à un serveur d'information d'administration (MIS : Management Information Server) suivant un protocole d'administration normalisé (CMIP) ou standard de fait (SNMP).

L'administration d'un réseau s'effectue à travers la manipulation des objets administrés dans le réseau. Un objet administré est l'abstraction d'une ressource du système.

Le système de supervision est l'ensemble des moyens qui permettent à un opérateur humain d'administrer le réseau local. Il est constitué d'un client et d'un ensemble d'agents (serveurs) répartis sur le réseau.

Le client présente une interface utilisateur à l'opérateur humain. Il est constitué d'un ensemble de tâches d'administration couvrant les trois fonctions d'administration présentées dans le paragraphe précédent. Il consulte un ou plusieurs agents pour obtenir les informations concernant le réseau. Le dialogue se fait par le protocole SNMP ou CMIP, en s'appuyant sur une infrastructure de communication (OpenView de HP, SunNet Manager de SUN, ...).

L'agent ATM possède toutes les informations et les moyens nécessaires pour administrer tout ou partie des ressources réelles du réseau. Plusieurs agents peuvent donc administrer des parties disjointes du réseau. Les objets sont gérés localement en les regroupant dans une base d'information (MIB : Management Information Base). L'agent est l'élément normalisé du système d'administration. Pour pouvoir échanger des messages avec les stations du réseau local ATM, l'agent est directement connecté sur une station du réseau. Il se comporte comme une application capable de communiquer via le réseau avec des applications distantes. L'agent ATM doit en particulier avoir accès aux entités de signalisation présentes sur la station.

III.2.4.3. Les particularités de l'administration de notre système

Du point de vue du protocole d'administration, les travaux sont plus avancés dans le monde TCP/IP (SNMP) que dans le monde OSI (CMIP). De plus, des MIB SNMP ont été définies pour la couche physique SONET/SDH [RFC1595] et pour la couche ATM [RFC1695]. C'est donc le protocole SNMP sur TCP/IP qui a été choisi pour la communication entre le client et les agents.

L'administration doit tenir compte des caractéristiques du système de communication qu'elle gère. L'architecture originale définie dans cette thèse implique la définition de nouveaux objets, ainsi qu'une gestion de la configuration adaptée (réseau ATM interne à la station).

La gestion de la configuration doit permettre à l'utilisateur de configurer le réseau suivant ses besoins, c'est à dire en fonction de ses applications. Ceci implique notamment de pouvoir établir des connexions logiques entre équipements et applications et de leur associer les paramètres adéquats. La configuration du réseau intervient lors de la mise en route du réseau (configuration initiale) et chaque fois qu'une nouvelle station se connecte sur le réseau (configuration partielle). La configuration initiale nécessite trois phases :

- La saisie de la description du réseau par l'administrateur, qui consiste à donner les caractéristiques des éléments du réseau et le schéma d'interconnexion.
- La construction de la représentation logique du réseau qui est mémorisée par le poste de configuration et la construction des informations de configuration qui sont envoyées aux entités de signalisation des stations et aux agents d'administration.
- La création des connexions nécessaires au fonctionnement du réseau, en particulier la méta-signalisation qui permet d'établir l'infrastructure de connexions nécessaire au protocole de signalisation.

Toutes ces phases dépendent étroitement du système de communication. Dans notre cas, la description des éléments du réseau nécessite des paramètres supplémentaires, comme le modèle de signalisation utilisé (UNI ou NNI), le nombre de cartes ATM de la station et les identifications de chacune d'elles, le nombre d'entrées/sorties vers le réseau ATM et les caractéristiques de chacune d'elles.

La deuxième fonction d'administration est la gestion des performances. Elle concerne la surveillance permanente de l'état du réseau par l'accès à des compteurs ou des variables. Elle permet de résumer à l'administrateur humain l'état de fonctionnement du réseau sous forme de statistiques, afin qu'il puisse améliorer les performances du réseau (éventuellement par une reconfiguration de tout ou partie des connexions ou par ajout d'un nouveau nœud de communication). Les objets relatifs à la gestion des performances sont définis dans les MIB de SONET/SDH et d'ATM. Les principales statistiques obtenues sont :

- Le débit de chacun des liens physiques du réseau ainsi que le débit encore disponible.
- Les taux d'erreur en émission et/ou réception au niveau de chaque interface physique.
- Les taux de perte de cellules ou de messages au niveau d'une interface ou au niveau d'une connexion.

Les objets manipulés permettent de satisfaire les contraintes de notre architecture. Par exemple, le réseau ATM interne à la station est administré avec les mêmes fonctions que le réseau externe.

III.2.4.4. Résumé

Ce paragraphe a présenté le modèle d'administration du réseau local ATM, basé sur le protocole SNMP. Les objets spécifiques à notre architecture originale, absents des bases d'information d'administration (MIB) définies par l'IETF, ont été énumérés.

Une des fonctions d'administration n'a pas été traitée dans ce paragraphe. Il s'agit de la détection des fautes. Le paragraphe suivant est consacré à cet aspect, ainsi qu'aux problèmes de redondance et de reconfiguration.

III.2.5. La gestion de la redondance et la reconfiguration

III.2.5.1. Introduction

La redondance permet d'assurer la sûreté de fonctionnement du réseau. Pour être activée, elle nécessite une détection des fautes. Le paragraphe III.2.5.2 y est consacré.

D'un autre côté, la reconfiguration permet de prévenir un écroulement des performances du réseau, en détournant certains flux sur des chemins moins encombrés. Le paragraphe III.2.5.3 présente cette fonction.

III.2.5.2. La gestion de fautes et la redondance

III.2.5.2.1. Introduction

La gestion de fautes consiste à détecter les problèmes de communication, au niveau d'une interface physique, au niveau d'une connexion particulière ou au niveau d'une interface logicielle (saturation mémoire par exemple).

Ces fautes peuvent donc intervenir à différents niveaux du modèle de référence. L'UIT-T a défini dans [I.610] les fonctions OAM (Operation And Maintenance) de gestion et de maintien d'un réseau large bande. La section III.2.5.1.2 présente ces fonctions ainsi que leur utilisation.

La section III.2.5.1.3 présente la gestion de la redondance que nous avons définie, service nécessaire particulièrement dans le cas des applications à fortes contraintes de disponibilité.

III.2.5.2.2. Les fonctions OAM

Les fonctions OAM relatives à la gestion de fautes appartiennent à trois classes :

- La détection de fautes concerne la détection et la prédiction de mauvais fonctionnement par des tests permanents ou périodiques. Cette catégorie génère donc des événements et alarmes.
- La protection du réseau concerne le cloisonnement des erreurs qui se produisent sur un réseau.
- L'information des fautes concerne les échanges d'informations relatives aux fautes détectées avec les autres entités d'administration.

De plus, les fonctions OAM sont classifiées en cinq niveaux hiérarchiques associés aux couches physique et ATM du modèle de référence ATM. A chaque niveau est associé un flux d'informations bidirectionnel. Ces niveaux sont les suivants :

- Le niveau connexion virtuelle (flux F5) concerne la surveillance de l'état des connexions (VCs).
- Le niveau chemin virtuel (flux F4) concerne la surveillance de l'état des chemins (VP).
- Le niveau de transmission physique (flux F3) concerne la surveillance des liens physiques.
- Le niveau section numérique (flux F2) concerne la surveillance de l'état des sections physiques (par exemple, trois sections STS3 à 51.84 Mbit/s dans un lien SONET STS3c à 155.52 Mbit/s).
- Le niveau de régénération (Flux F1) concerne la surveillance de parties des sections numériques.

Les flux F1, F2 et F3 correspondent à la couche physique et permettent de diagnostiquer l'état des liens physiques reliant les nœuds du système. Dans le cas d'une couche physique SONET, les flux F1 et F2 sont véhiculés dans la partie SOH (Section OverHead) de la trame, le flux F3 étant véhiculé dans la partie POH (Path OverHead).

Les flux F4 et F5 correspondent à la couche ATM et permettent de diagnostiquer l'état des chemins virtuels et des connexions virtuelles reliant les applications/utilisateurs sur le réseau. Ils sont échangés à l'aide de cellules ATM sur les connexions (VPC et VCC) ouvertes afin de tester ces connexions de bout en bout en empruntant le même chemin que les cellules correspondant au trafic des applications. Les cellules OAM sont insérées dans le flux de cellules et extraites du flux par l'entité d'administration afin qu'elles ne soient pas vues des applications.

Le codage des cellules OAM, les valeurs réservées de VPI/VCI pour les cellules OAM ainsi qu'une liste des types de messages sont spécifiés dans [I.610].

III.2.5.2.3. La redondance

La redondance des connexions est un service requis par de nombreuses applications distribuées temps réel et plus généralement par les applications à fortes contraintes de disponibilité. Il s'agit donc de basculer le trafic de chaque connexion concernée par ce service sur sa connexion de secours lorsqu'une faute est détectée sur le réseau. La connexion de secours devra, lorsque c'est possible, emprunter un chemin différent de la

connexion normale. Ceci requiert l'analyse et la reconstitution du chemin emprunté par la connexion normale.

La création d'un chemin est habituellement effectuée de manière répartie, ce qui ne permet pas de garantir qu'une connexion de secours empruntera un chemin différent de la connexion normale associée. La réalisation d'un service de création centralisé de chemins comporte deux étapes :

- Le calcul d'un chemin qui pourra être fait selon différents critères en fonction de l'objectif à atteindre. Le critère pourra être l'équilibre des débits sur le réseau si ce service est requis par la gestion des performances. Si ce service est requis par la gestion de fautes et de la redondance, le critère sera la différence maximale entre chemin normal et chemin calculé.
- L'ouverture du chemin proprement dite.

Il ne s'agit pas ici de reconfiguration mais de redondance car la connexion de secours est ouverte en général en même temps que la connexion normale. En effet, le basculement doit être le plus rapide possible ce qui n'est pas compatible avec l'ouverture d'une connexion, en particulier si le chemin doit être calculé de façon globale et s'il faut avertir toutes les entités de signalisation sur ce chemin avant d'envoyer le Setup/Msetup.

Le basculement peut à priori être géré à trois niveaux :

- Au niveau ATM, c'est à dire directement dans les tables de commutation. Ceci s'avère difficile, d'autant plus qu'il n'y a aucune garantie sur l'arrivée dans le bon ordre des cellules après un basculement.
- Au niveau AAL-SSCS, c'est à dire à un niveau où des messages sont manipulés. Cette solution est détaillée dans la suite.
- Au niveau des applications. Cette solution n'est guère envisageable. Le programmeur n'a pas à gérer lui-même la redondance des connexions ATM, qui est un service de niveau inférieur. Par contre, c'est à lui de gérer la redondance d'application, qui est un problème complètement différent.

La gestion du basculement au niveau SSCS a été choisi. Ceci permet de rendre transparent ce service pour les applications. De plus, le basculement se fait après la fin d'émission d'un PDU complet, ce qui permet d'éviter les problèmes de déséquilibrage. Lors de l'ouverture d'une connexion, l'application spécifie dans les paramètres de qualité de service si elle désire une connexion classique ou une connexion redondante. Dans ce dernier cas, deux connexions sont ouvertes, en utilisant des chemins différents. Lors de la

détection d'une défaillance sur la connexion en cours d'utilisation, le module de gestion de la redondance est averti par l'administration qu'il est nécessaire de basculer sur la connexion de secours. Le basculement peut s'opérer de deux manières :

- Au niveau du destinataire si tous les messages sont envoyés simultanément sur les deux connexions.
- Au niveau de l'émetteur si une seule connexion est effectivement utilisée à un instant donné. Dans ce cas, il faut aussi un basculement au niveau du destinataire puisque ce n'est plus la même connexion ATM qui est utilisée en réception.

Bien que les ressources soient réservées sur le réseau, la première solution occupe effectivement la bande passante, alors que celle-ci pourrait être utilisée par le trafic non prioritaire en vertu de la politique définie dans le paragraphe III.2.1. Néanmoins, elle permet de diminuer le délai de basculement. Elle est surtout plus simple à gérer puisqu'il suffit, au niveau du récepteur, d'associer les deux connexions ATM au même point d'accès. Seules les données de la connexion active sont traitées, les autres sont jetées. Le basculement se fait sur ordre de l'administration, en inversant le rôle des deux connexions.

La deuxième solution nécessite un traitement particulier à la source. Sur indication de l'administration, il faut changer l'association entre le point d'accès AAL et le port de communication utilisé par l'application, c'est à dire remplacer le point d'accès de la connexion défaillante par celui de la connexion de secours. Pour éviter les problèmes de synchronisation entre les basculements de l'émetteur et du récepteur, il suffit, au niveau du récepteur, d'associer les deux connexions ATM au même point d'accès. Le basculement se fait alors automatiquement, dès qu'un message est reçu sur la connexion de secours.

La gestion de la redondance est le service de plus bas niveau dans la couche SSCS. Ceci permet en particulier de laisser aux services supérieurs le traitement des problèmes de resynchronisation ou de réémission. En effet, lors du basculement, des parties de messages peuvent être perdues ou être délivrées en double, en fonction des chemins des deux connexions. Cette perturbation passagère est rétablie par les services de niveau supérieur, si le transfert est en mode assuré. Dans un transfert non assuré, la perte de données n'est pas critique.

Ainsi, la gestion de la redondance est complètement définie. Le choix d'une des deux solutions permet de privilégier soit le délai de basculement et la facilité d'implémentation, soit la libération de bande passante sur le réseau.

III.2.5.3. La reconfiguration

La reconfiguration partielle du réseau peut s'avérer nécessaire lorsque les taux de perte ou d'erreur sont trop élevés en certains points du réseau. Ce besoin peut aussi survenir lorsque le réseau est mal équilibré, c'est-à-dire lorsque des liens physiques sont très utilisés alors que d'autres le sont moins.

Dans le premier cas, un agent signale à l'administrateur le franchissement d'un seuil par la ou les variables relatives aux taux d'erreur et de perte. Le seuil dépend de la valeur des paramètres de qualité de service associés aux connexions.

L'administrateur prend alors la décision de fermer les connexions concernées et en avertit l'agent. Les fermetures seront effectuées par les entités de signalisation (SIG) qui gèrent la station dont les qualités de service des connexions ne sont plus respectées. De plus, l'administrateur demande la réouverture des connexions aux SIG des initiateurs (ouverture par une tierce partie).

Dans le deuxième cas, l'administrateur qui surveille continuellement les valeurs des débits disponibles sur chacun des liens physiques du réseau, doit prendre la décision de reconfigurer une partie des connexions qui passent par les chemins engorgés. Les nouvelles connexions devront être réparties sur d'autres chemins. Le "meilleur" chemin est donc déterminé au niveau de l'administrateur (qui est le seul à avoir une vision globale du réseau) puis envoyé à l'agent qui gère la station où se trouve l'initiateur de la connexion à recréer. Pour cela, il existe deux possibilités :

- Ajouter un message d'ouverture de connexion qui donne le chemin complet que devra suivre la connexion. Cette méthode risque de poser des problèmes de compatibilité avec la norme de signalisation.
- Avertir les SIG sur le chemin que devra prendre la connexion du lien physique à utiliser pour ouvrir cette connexion lorsque le Setup/Msetup arrivera. Les SIG sont averties par l'intermédiaire des agents associés. Cette méthode ne remet pas en cause la norme de signalisation.

Cette méthode de création d'un chemin de façon globale permettrait d'équilibrer la charge du réseau. Une telle technique présente un grand intérêt dans le cas des applications distribuées temps réel dont 90% des connexions sont connues au démarrage du réseau.

III.2.5.4. Résumé

Ce paragraphe a présenté la gestion de fautes dans un réseau ATM. Puis une méthode de gestion de la redondance et de basculement a été décrite. Pour le basculement, deux solutions ont été proposées, l'une privilégiant le délai de basculement et la facilité d'implémentation, l'autre la bande passante disponible sur le réseau. Enfin, les problèmes de reconfiguration, distincts de la redondance, ont été abordés.

III.2.6. Conclusion

Cette partie a présenté les **fonctions de gestion des communications entre les équipements**, afin de compléter l'architecture du système de communication.

Une méthode de contrôle de congestion a été présentée. Elle permet de borner le délai d'acheminement des messages temps réel, ce qui est indispensable pour le domaine d'application visé. Elle est basée sur la limitation du nombre de chemins et de la longueur des rafales. Elle tire profit des caractéristiques des composants utilisés au niveau ATM et AAL.

Pour la transmission des données sans connexion, les protocoles en cours de définition que sont IP sur ATM et le LAN Emulation apportent une solution. Bien que celle-ci soit lourde, elle ne gêne pas l'écoulement du trafic temps réel.

La signalisation est particulière puisque notre système de communication contient un commutateur ATM embarqué. Il est donc nécessaire de supporter à la fois les protocoles UNI et NNI, comme le fait un commutateur ATM classique. Un protocole d'ouverture de connexions par une tierce partie a été ajouté afin de répondre au besoin des applications de supervision.

L'administration du système de communication est basée sur le protocole SNMP. Les objets nécessaires à la gestion de notre architecture particulière ont été ajoutés.

Enfin, des méthodes pour la gestion de la redondance et la reconfiguration ont été proposées.

III.3. Conclusion

Ce chapitre a présenté le système de communication que nous avons défini. Dans un premier temps, une **architecture de couplage des équipements originale** a été proposée avec notamment un *commutateur ATM embarqué*, un *micro-contrôleur local* et une *organisation protocolaire* particulière. Ces trois éléments permettent entre autres de connecter plusieurs équipements audio/vidéo sur une machine standard et de construire des topologies variées. De plus, des services temps réel de haut niveau ont été sélectionnés pour profiter d'une implantation au plus près du réseau sur le micro-contrôleur, sous la forme d'un télé-DMA. Ces services sont une option du système de communication.

Ensuite, les fonctions nécessaires à la **communication entre les machines** ont été décrites. Les caractéristiques particulières de l'architecture ont conduit à la définition d'opérations nouvelles, comme un *contrôle du trafic* pour garantir l'absence de congestion sur le réseau, un mécanisme d'*ouverture de connexions par une tierce partie* et une *gestion de la redondance*.

Maintenant que le système de communication est entièrement défini, il convient d'évaluer les performances de celui-ci. Le chapitre suivant y est consacré.

Chapitre IV

L'évaluation des performances

Les caractéristiques de l'architecture du système de communication ayant été présentée, il convient maintenant de connaître les performances de celle-ci. Ce chapitre présente dans un premier temps les objectifs recherchés dans l'évaluation des performances. Les raisons du choix du langage VHDL et les caractéristiques de celui-ci pour l'évaluation des performances sont ensuite abordées. Puis, le modèle utilisé est décrit. Finalement, ce chapitre compare les performances de l'architecture du système définie au cours de cette thèse avec celles des systèmes actuels.

IV.1. Les objectifs

Ce paragraphe présente les résultats attendus du modèle de l'architecture du système de communication. Ces résultats sont de plusieurs ordres :

- Comparer les performances du traitement des flux temps réel dans différentes implantations. La méthode décrite dans le chapitre précédent (transferts critiques) est analysée par rapport à des implantations sur TCP/IP (la méthode classique proposée pour exécuter les applications au-dessus du réseau ATM (IP sur ATM)), dans un fonctionnement avec ou sans le micro-contrôleur local. Les comparaisons sont effectuées du point de vue des applications, c'est à dire sur le délai de bout en bout entre applications distantes (émission + réception).
- Mesurer le comportement du système dans différentes situations, normales ou exceptionnelles. En particulier, le modèle doit permettre d'estimer le débit maximum supporté et les goulots d'étranglement de l'architecture. Ces performances permettront aussi de comparer l'architecture à des systèmes existants, dans le cas de transferts d'informations classiques.

Un modèle analytique a été défini dans un premier temps. Celui-ci a permis de calculer les délais d'acheminement des messages entre applications pour une distribution de longueurs, mais pour une seule connexion ATM à la fois. Cependant, il devient trop complexe lorsque l'on veut estimer le comportement du système avec plusieurs messages transférés simultanément. De plus, il est très difficile d'introduire la charge des processeurs et des bus. Par conséquent, une simulation du système s'est avérée nécessaire. Le langage VHDL [Air90] a été choisi pour décrire le modèle. Le paragraphe suivant présente les caractéristiques de VHDL et ces avantages pour la simulation des systèmes.

IV.2. Le langage VHDL et la simulation des systèmes

IV.2.1. Présentation de VHDL

VHDL ((VHSIC : Very High Speed Integrated Circuits) Hardware Description Language) est, comme son nom l'indique, un langage de description d'architectures, normalisé au niveau international par l'IEEE [IEE87]. Il permet de décrire des systèmes matériels (par opposition à logiciels) à un haut niveau d'abstraction. A partir de celui-ci, de nombreux outils permettent de synthétiser des circuits intégrés en silicium. Regardons plus en détail les caractéristiques de ce langage.

VHDL permet d'exprimer l'organisation d'un système. La fonction du système est spécifiée par une fonction de transfert des entrées vers les sorties. Le temps fait explicitement partie de cette transformation. L'organisation du système est exprimée en termes de composants interconnectés, constitués d'un ensemble de modèles et d'algorithmes utilisés par ces modèles.

Un modèle peut être imaginé en deux parties : une vue externe qui montre ses connexions avec le monde extérieur et une vue interne qui décrit sa réalisation sous forme d'interconnexions d'autres modèles plus simples ou sous forme d'algorithmes. La vue externe se nomme *entité*, et la vue interne est appelée *architecture*. Une même entité peut avoir plusieurs architectures associées. Le choix de l'architecture se fait avec une déclaration de *configuration*. Ceci permet d'exprimer le plus souvent divers algorithmes ou divers degrés de raffinement de la vue interne.

De plus, il est possible de définir des modules pour les utiliser dans les corps des architectures. Ces procédures ou fonctions sont classées dans des paquets appelés *paquetages* (packages en anglais). Ceci permet à plusieurs unités d'utiliser des ressources partageant la même déclaration. Ces paquetages sont aussi découpés en deux parties : leur vue externe, c'est à dire l'aperçu de toutes les possibilités qu'ils exportent, est découplée de la vue interne qui contient l'algorithmique réalisant ces fonctionnalités. La vue externe est appelée *spécification du paquetage* et la vue interne *corps du paquetage*. Les paquetages contiennent aussi des déclarations de types, de signaux globaux ou de composants.

VHDL permet de décrire un système matériel dans trois styles différents :

- Description *comportementale* : description sous forme de programmes séquentiels à la manière d'un langage de programmation de haut niveau tel Pascal, C, ...
- Description *flot de données* : description au niveau des signaux passant au travers des portes.

- Description *structurelle* : description de la structure en terme de composants interconnectés.

Il faut noter que ces trois types de descriptions peuvent être mélangés à l'intérieur d'une même architecture.

VHDL est un langage travaillant sur des processus concurrents réagissant sur des événements. Une spécification d'architecture est composée d'un ou plusieurs processus qui interagissent au moyen de signaux. Les processus VHDL sont cycliques infiniment et sont synchronisés sur des événements. Pour cela, VHDL gère la notion de temps. Aussi, il y a deux sortes d'instructions : les instructions séquentielles à l'intérieur d'un processus et les instructions concurrentes.

D'autre part, VHDL est un langage typé : toute constante, toute variable, tout signal doit avoir un type avant d'être utilisé. On trouve bien sûr les types simples classiques (entier, flottant, booléen, caractère, ...) mais aussi des types composés tels que les tableaux, les enregistrements, les pointeurs, les types énumérés et les fichiers.

IV.2.2. L'utilisation de VHDL pour l'évaluation de performances

Contrairement à son appellation, VHDL s'élève bien au-dessus des circuits intégrés pour s'appliquer aux cartes de composants et même à des systèmes entiers comprenant à la fois des parties matérielles et logicielles. Il permet ainsi de simuler des systèmes complexes, en exprimant leur organisation en termes de composants interconnectés. VHDL est bien adapté à la modélisation des systèmes et à l'évaluation de leurs performances pour plusieurs raisons :

- Le temps fait explicitement partie de la description.
- Le fonctionnement sur des processus et des événements concurrents est adéquat pour exprimer le comportement des systèmes de communication qui sont par nature parallèles.
- Les notions d'architecture et de configuration permettent d'affiner le modèle au fur et à mesure, sans remettre en cause le reste de la description. Il est aussi possible de construire différents scénarios en utilisant la même base de description.
- Le pouvoir algorithmique du langage et son typage offrent une grande souplesse pour exprimer le comportement du système.

L'environnement de simulation accompagnant le langage VHDL est le dernier aspect important. Il rend aisée cette phase et offre des outils pour surveiller le

comportement du système, en particulier pour l'évolution des signaux et l'indication des situations exceptionnelles.

Enfin, le mixage de différents niveaux de descriptions permet d'intégrer au modèle des circuits spécifiques de l'architecture complètement spécifiés en VHDL.

Toutes ces caractéristiques nous ont conduit à décrire le modèle de l'architecture du système en VHDL. Le paragraphe suivant présente ce modèle.

IV.3. Le modèle

Le modèle VHDL doit répondre aux objectifs présentés dans le premier paragraphe. Il est composé de deux entités principales (cf. figure IV.1) :

- La station qui simule la carte principale définie précédemment.
- L'analyseur qui permet de calculer les performances.

La station est décomposée en cinq sous-entités : la machine hôte, le micro-contrôleur local, la segmentation et le réassemblage des cellules, l'interface physique et les applications. A chaque niveau, des temps de traitement sont affectés et tiennent compte de la charge du système. Ces temps sont tirés des documentations des composants utilisés ou d'évaluations.

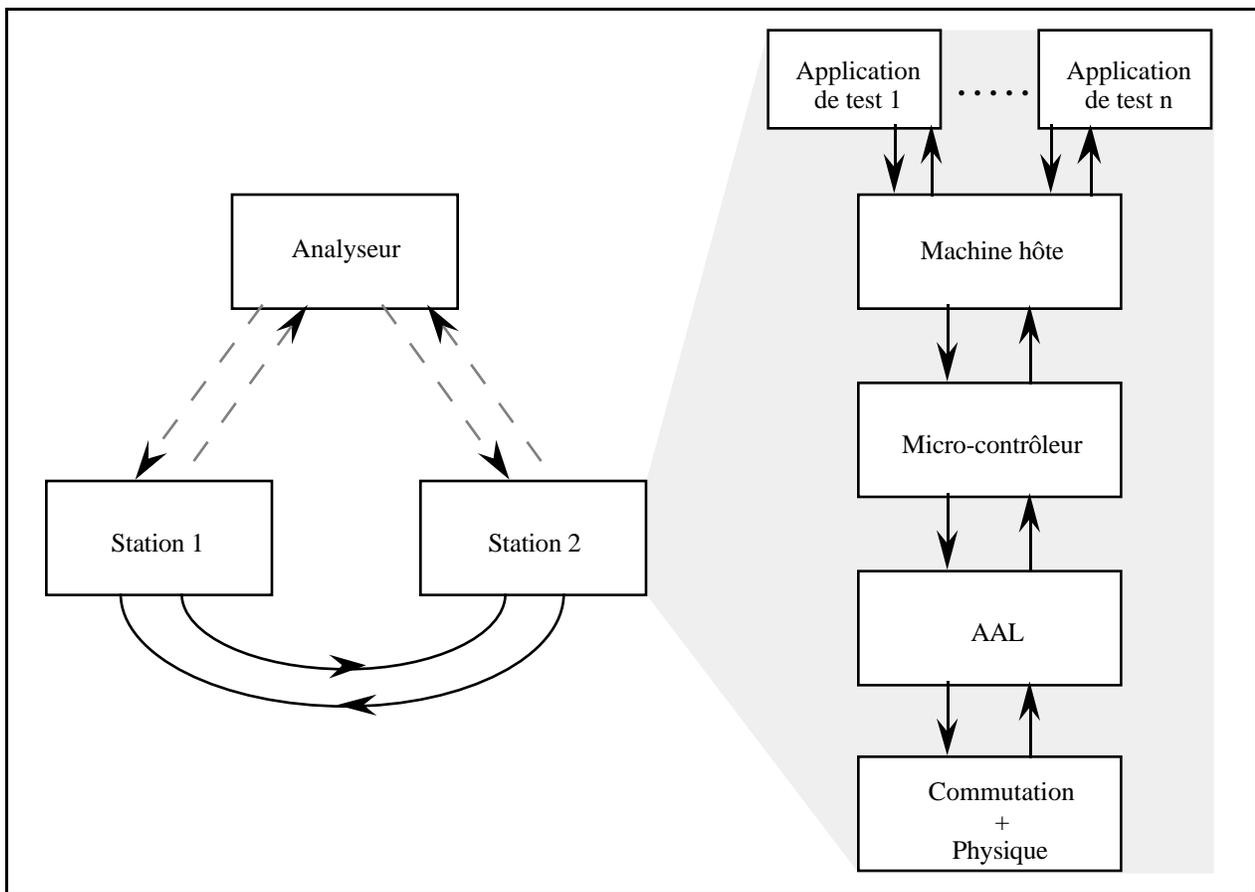


Figure IV.1 : Le modèle du système de communication

L'entité "machine hôte" simule le processeur hôte et les accès au bus fond de panier. Elle prend en compte l'exécution des protocoles et les applications. L'entité "micro-contrôleur" est la plus complexe. Elle gère les conflits d'accès au bus de la carte réseau et la charge du processeur embarqué. Elle s'occupe aussi des transferts entre le tampon de segmentation et de réassemblage et le tampon d'émission et de réception, et entre ce dernier et la machine hôte. L'entité "AAL" se charge de la segmentation et du réassemblage des messages, et du contrôle de flux en émission suivant le débit de chaque connexion. L'interface au réseau, c'est à dire la commutation et l'interface physique sont prises en compte dans le module "Commutation + Physique". Les entités "Application de test" permettent de générer les messages à envoyer, en fonction du type de performance recherché. Un module réseau pourra s'ajouter au modèle afin de connecter plusieurs stations, et d'évaluer l'incidence du réseau sur le comportement du système. Le modèle est décrit en VHDL au niveau comportemental, afin de ne pas compliquer inutilement la description et garder des temps de simulation raisonnables.

A l'intérieur de la station, les **actions** suivantes ont été considérées pour évaluer le temps de transmission entre applications distantes :

- Le transfert entre la mémoire de l'application sur la machine hôte et le tampon de transmission (respectivement réception) sur l'interface réseau (T_{m1} pour un mot de 32 bits et T_{d1} pour l'initialisation du DMA).
- Le pilote de l'interface réseau (T_p).
- La gestion des tampons d'émission et de réception (T_g).
- Les mouvements de données entre le tampon d'émission (respectivement réception) et le tampon de segmentation (respectivement réassemblage) sur la carte d'interface, en fonction de la taille des blocs (T_{m2} pour un mot de 32 bits et T_{d2} pour l'initialisation du DMA).
- Le traitement des couches câblées AAL, ATM et Physique, incluant le temps d'attente avant émission au niveau de la segmentation en fonction du débit (T_{atm}).
- L'exécution des protocoles spécifiques suivant le type de performance recherché (TCP/IP, transport rapide sur AAL, traitement optimisé des flux temps réel) en séparant l'émission et la réception, aussi bien sur la machine hôte que sur le processeur embarqué.

Deux types d'entrées permettent de configurer le modèle afin de définir le type de simulation désiré :

- Les paramètres relatifs à l'architecture du système :
 - La taille des blocs transférés entre le tampon de segmentation et de réassemblage et le tampon d'émission et de réception.
 - La puissance du processeur hôte.
 - Le temps de traitement du pilote réseau.
 - Le temps de traitement des protocoles.
- Les paramètres relatifs au trafic :
 - Le type de trafic (TCP/IP, transport direct sur l'AAL, variables de contrôle temps réel, mode assuré).
 - Le débit.
 - La longueur des messages.
 - La distribution des émissions.
 - La charge des processeurs, par exemple les applications s'exécutant sur le processeur hôte.

Dans l'état actuel du modèle, deux **hypothèses** sont en vigueur. Elles ne sont pas gênantes pour les évaluations que nous voulons faire ici. Elles pourront être levées en affinant le modèle, en particulier la première :

- Il n'y a pas d'attente dans les commutateurs.
- L'établissement des connexions est effectuée au préalable.

Les **caractéristiques** du système pour les comparaisons présentées dans le paragraphe suivant sont :

- Un temps d'accès mémoire de 3 cycles à 33 MHz en lecture ou en écriture, soit $T_{m1} = T_{m2} = 90,9$ ns.
- Une puissance du processeur hôte égale à 50 MIPS, sauf précision contraire.
- Une puissance du processeur embarqué égale à 30 MIPS (fixé par l'architecture).
- Un débit du réseau de 155,52 Mbit/s, soit $T_{cell} = 2,72$ μ s.
- Une taille des blocs transférés entre le tampon d'émission et de réception et le tampon de segmentation et de réassemblage égale à 10 cellules, soit 480 octets.

Les puissances des processeurs données ici correspondent aux puissances crêtes. Le modèle VHDL des processeurs permet de tenir compte de la puissance réelle délivrée par les processeurs, en fonction des transferts sur le bus et des accès mémoire. Les deux premières caractéristiques correspondent aux valeurs typiques que l'on trouve aujourd'hui pour des équipements de bonne qualité. D'un côté, l'évolution des temps d'accès mémoire est très lente. Nous garderons donc cette valeur tout au long des évaluations. Par contre, la puissance des processeurs augmente très vite et des éléments de plusieurs centaines de MIPS existent déjà. Par conséquent, nous étudierons à la fin de ce chapitre l'évolution des performances de notre système de communication en fonction de l'augmentation de la puissance des processeurs.

IV.4. La comparaison des performances

IV.4.1. Les éléments à comparer

Le but de ce paragraphe est de comparer les performances du système de communication dans quatre modes de fonctionnement :

- La configuration sans le micro-contrôleur local, avec un transfert directement sur l'AAL. Elle sera nommée dans la suite AAL.

- La configuration sans le micro-contrôleur local, avec TCP/IP. C'est le mode de fonctionnement utilisé dans la plupart des systèmes actuels. Il sera nommé dans la suite TCPIP.
- La configuration avec le micro-contrôleur local, avec les services optimisés de transfert critique définis précédemment. Elle sera nommée dans la suite AAL_MC.
- La configuration avec le micro-contrôleur local, avec TCP/IP. Elle sera nommée dans la suite TCPIP_MC.

La figure IV.2 montre l'organisation des protocoles dans les différentes configurations à comparer, avec leur répartition entre le processeur hôte et le processeur embarqué pour les deux dernières configurations.

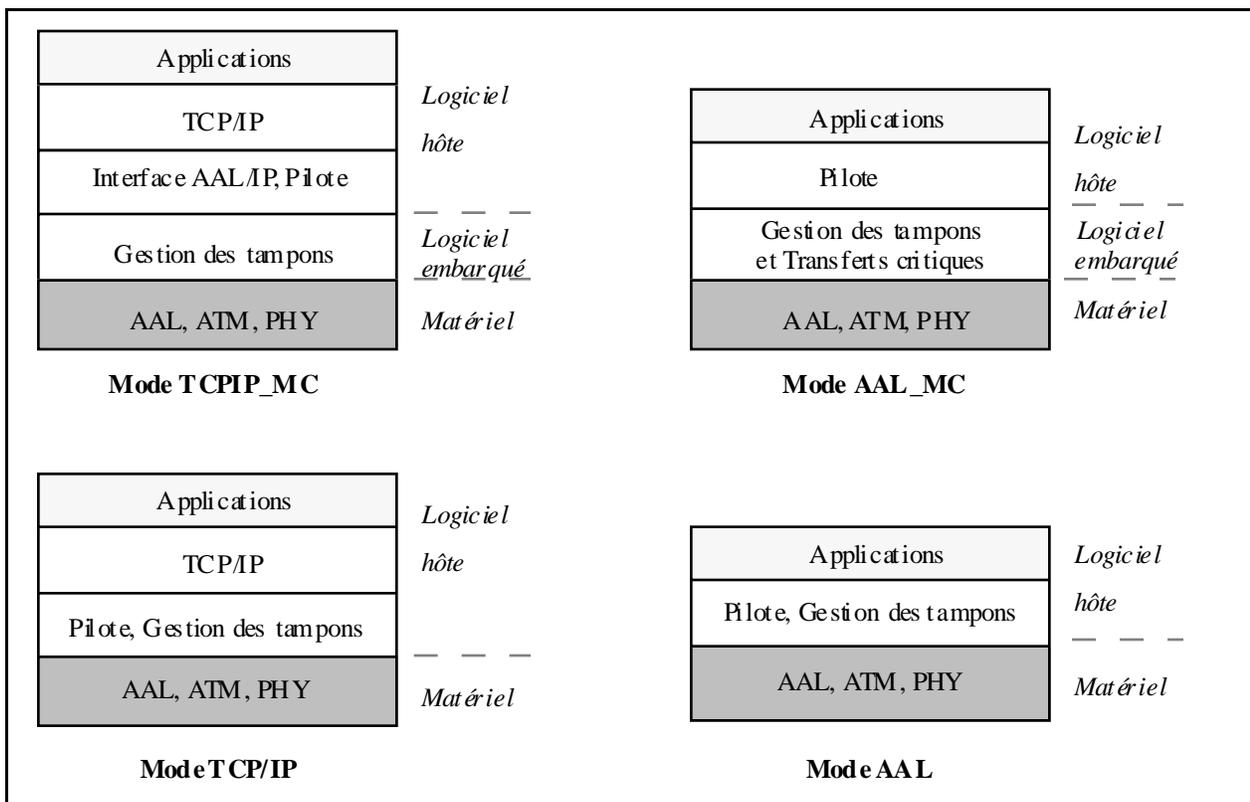


Figure IV.2 : Les implantations à comparer

Pour mieux comprendre le modèle, regardons comment s'exprime le délai subit entre l'émetteur et le récepteur par un message de longueur L octets. Dans le cas d'un envoi sur TCP/IP, le délai s'exprime sous la forme suivante (en utilisant les notations définies dans le paragraphe précédent) :

$$D = (T_{atm} + \left[\frac{L}{4}\right] * T_{m1} + T_{d1} + \left[\frac{L+H}{4}\right] * T_{m2} + T_{d2} + T_p + T_g) * 2 + T_{tcpip}$$

où T_{tcpip} est le temps de traitement du protocole TCP/IP en émission et en réception, H la longueur des en-têtes ajoutés par le protocole TCP/IP, et $[x]$ la partie entière supérieure de la valeur x . Pour le protocole TCP/IP, les évaluations des temps de traitement sont tirés de mesures de performances de différentes implémentations efficaces du protocole. Elles sont similaires à celles fournies dans [Kay93]. Ces temps incluent les temps des recopies de tampons effectuées par le protocole TCP/IP.

Développons le temps de traitement T_{atm} . Le traitement des trois couches AAL, ATM et PHY est effectué en pipeline sur les cellules, chaque étage coûtant un temps cellule (T_{cell}). Cependant, le temps d'attente d'une cellule dans l'AAL est fonction du débit. S'il n'y a pas de congestion, ce temps est égale à $\left\lceil \frac{155,52}{\text{débit}} \right\rceil * T_{cell}$. Ainsi,

$$T_{atm} = \left(\left\lceil \frac{155,52}{\text{débit}} \right\rceil * \text{nombre_de_cellules} + 2 \right) * T_{cell}$$

Si Q est le nombre d'octets ajoutés par le protocole AAL 5, nous avons donc pour le message de longueur L au niveau de l'application un nombre de cellules égal à :

$$\text{nombre_de_cellules} = \left\lceil \frac{L+H+Q}{48} \right\rceil$$

D'où l'expression complète suivante pour le transfert d'un message sur TCP/IP :

$$\left(\left(\left\lceil \frac{155,52}{\text{débit}} \right\rceil * \left\lceil \frac{L+H+Q}{48} \right\rceil + 2 \right) * T_{cell} + \left\lceil \frac{L}{4} \right\rceil * T_{m1} + T_{d1} + \left\lceil \frac{L+H}{4} \right\rceil * T_{m2} + T_{d2} + T_p + T_g \right) * 2 + T_{tcpip}$$

Dans le cas du transfert d'un message avec un protocole de transport direct sur l'AAL, le délai d'émission et réception est le suivant :

$$\left(\left(\left\lceil \frac{155,52}{\text{débit}} \right\rceil * \left\lceil \frac{L+Q}{48} \right\rceil + 2 \right) * T_{cell} + \left\lceil \frac{L}{4} \right\rceil * T_{m1} + T_{d1} + \left\lceil \frac{L}{4} \right\rceil * T_{m2} + T_{d2} + T_p + T_g \right) * 2 + T_{tr}$$

où T_{tr} est le temps de traitement du protocole au niveau du Transputer.

Ces équations sont valables uniquement en l'absence de conflits sur les bus et de charge des processeurs, et avec un seul message à la fois. Ceci est évidemment une situation irréaliste. Le modèle VHDL permet d'ajouter ces contraintes aux équations ci-

dessus. Les résultats présentés dans la suite de ce chapitre sont issus de simulations effectuées à partir de ce modèle.

IV.4.2. Les performances du système pour l'envoi de messages classiques

Ce paragraphe compare les performances de l'envoi de messages dans les différentes configurations possibles, en terme de délai de bout en bout et de débit maximum supporté, au niveau application. Il présente d'abord les valeurs optimales de ces paramètres, avec une seule connexion en émission, puis une connexion en émission et une en réception. Ensuite, l'évolution des paramètres en fonction du nombre de connexions simultanées est étudiée. Toutes ces évaluations sont réalisées pour une distribution des longueurs des messages, allant de 32 à 16384 octets (32, 64, 128, 256, 512, 1024, 2048, 4096, 8192 et 16384 octets). Pour toutes les courbes présentées dans la suite, une échelle logarithmique est utilisée pour l'axe supportant les longueurs.

Dans toutes les évaluations suivantes, le débit alloué par connexion est le débit maximum divisé par le nombre de connexions, chaque connexion ayant le même débit. Ainsi, pour une seule connexion en émission, ou une en émission et une en réception, le débit alloué pour chacune d'elles est de 155,52 Mbit/s (au niveau réseau).

Avec un réseau ATM à 155,52 Mbit/s, le débit maximum théorique au niveau ATM est de 140,85 Mbit/s, en raison des 5 octets d'en-tête ATM. Au niveau de l'AAL 5, le débit maximum théorique est fonction de la longueur du message puisque 8 octets de champs d'information sont ajoutés à chaque message. Ainsi, un message de 32 octets ne peut dépasser les 93,90 Mbit/s, un message de 128 octets les 125,20 Mbit/s alors que pour des tailles supérieures, le débit maximum théorique tend vers 140,85 Mbit/s (cf. figure IV.3). Les sauts de la courbe sont dues au taux de remplissage de la cellule par les données.

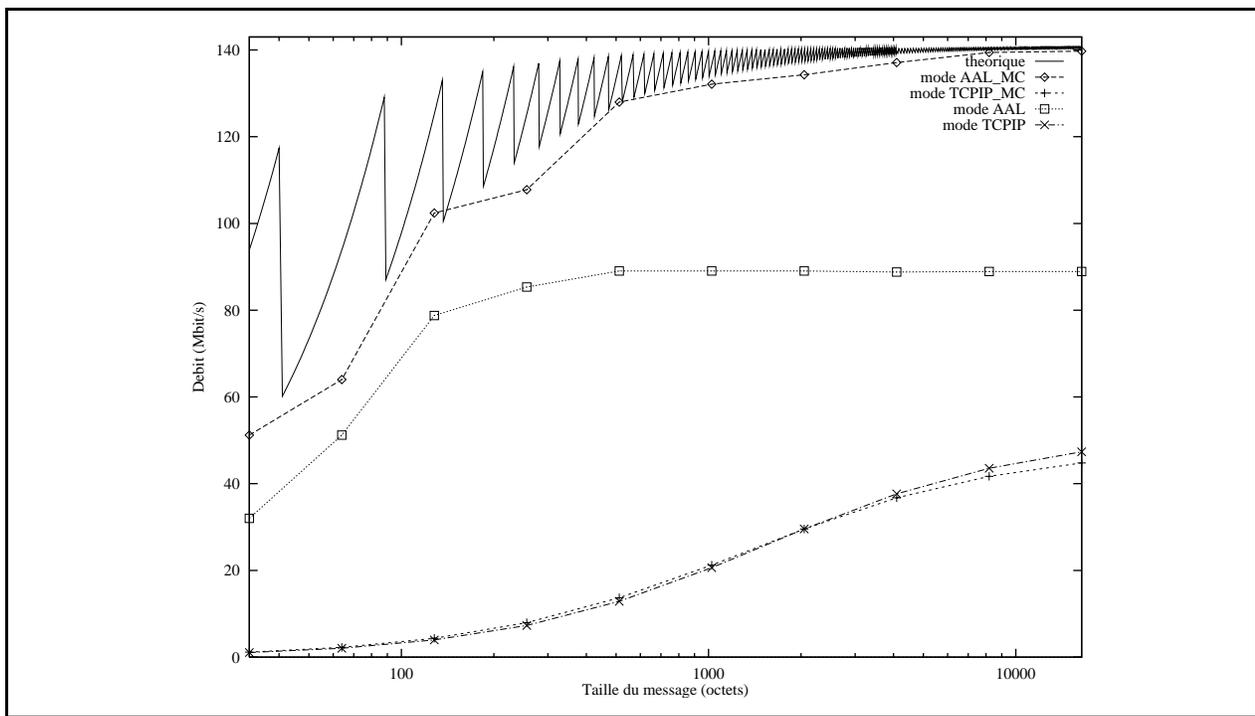


Figure IV.3 : Le débit maximum pour une connexion en émission

La figure IV.3 donne le débit maximum au niveau application supporté par le système de communication, pour les quatre modes. L'évaluation est faite avec une seule station émettant sur le réseau et une station recevant les messages, la connexion ATM ayant un débit de 155,52 Mbit/s. On voit que le débit atteint dans le mode AAL_MC est proche du débit théorique. Pour des messages de petite taille, l'écart est assez important en raison des coûts de traitement indépendant de la longueur des messages, comme l'initialisation du DMA ou la prise en compte de l'envoi ou de la réception d'un message. Les irrégularités de la courbe sont dues au choix des valeurs discrètes. Si toutes les valeurs avaient été considérées, nous aurions obtenu une courbe similaire à la courbe théorique. Le débit pour le mode AAL sans micro-contrôleur est limité par la puissance du processeur. Il en résulte un fléchissement pour les messages de taille supérieure à 256 octets.

Outre le débit maximum, il est intéressant de connaître le délai de transmission de bout en bout au niveau application. Les figures IV.4a et IV.4b présentent ce délai, toujours pour les quatre modes. Pour le mode AAL_MC, le délai est inférieur à 100 μ s pour des messages de 512 octets maximum. Pour des messages de taille supérieure à 512 octets, une augmentation plus importante du délai est observée dans le mode AAL (cf. figure IV.4b). Cet effet résulte du choix de la taille des blocs transférés entre le tampon d'émission et de réception et le tampon de segmentation et de réassemblage qui est de 10 cellules, soit 480 octets. Pour une telle taille de message, il faut effectuer deux transferts entre ces tampons.

Les figures IV.3, IV.4a et IV.4b, mêmes si elles ne représentent pas une situation réelle, permettent de contrôler la validité du modèle. Pour TCP/IP, les délais obtenus sont similaires à [Lin95, Wol93], voire même globalement inférieurs, et montrent ainsi que la base des comparaisons est correcte.

Par contre, les débits mesurés dans [Lin95] avec la carte FORE en AAL 5 [For94] sont très inférieurs à ceux obtenus avec notre système. En effet, le débit maximum obtenu pour un message de 8 Ko est de 30 Mbit/s alors que dans notre cas il est quatre fois plus élevé. Le délai est lui aussi inférieur puisqu'il est de 7 ms pour un message de 8 Ko avec la carte FORE, contre 1,1 ms dans notre cas.

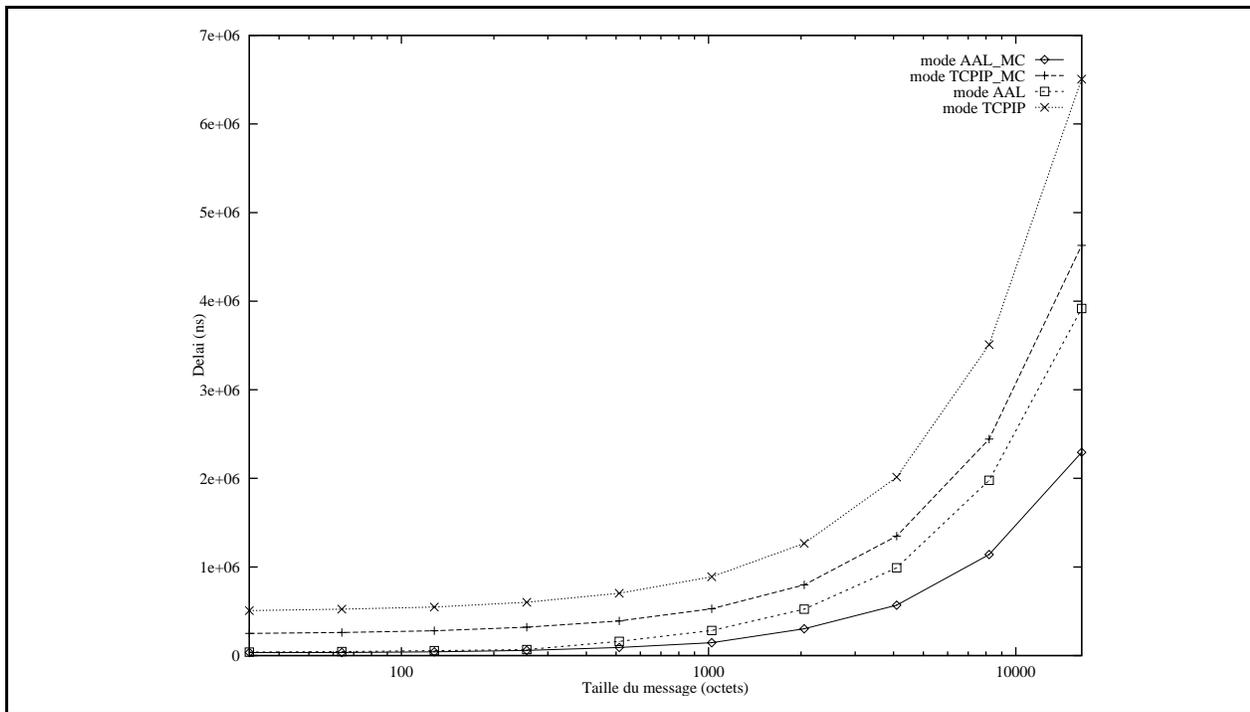


Figure IV.4a : Le délai de bout en bout pour une connexion en émission

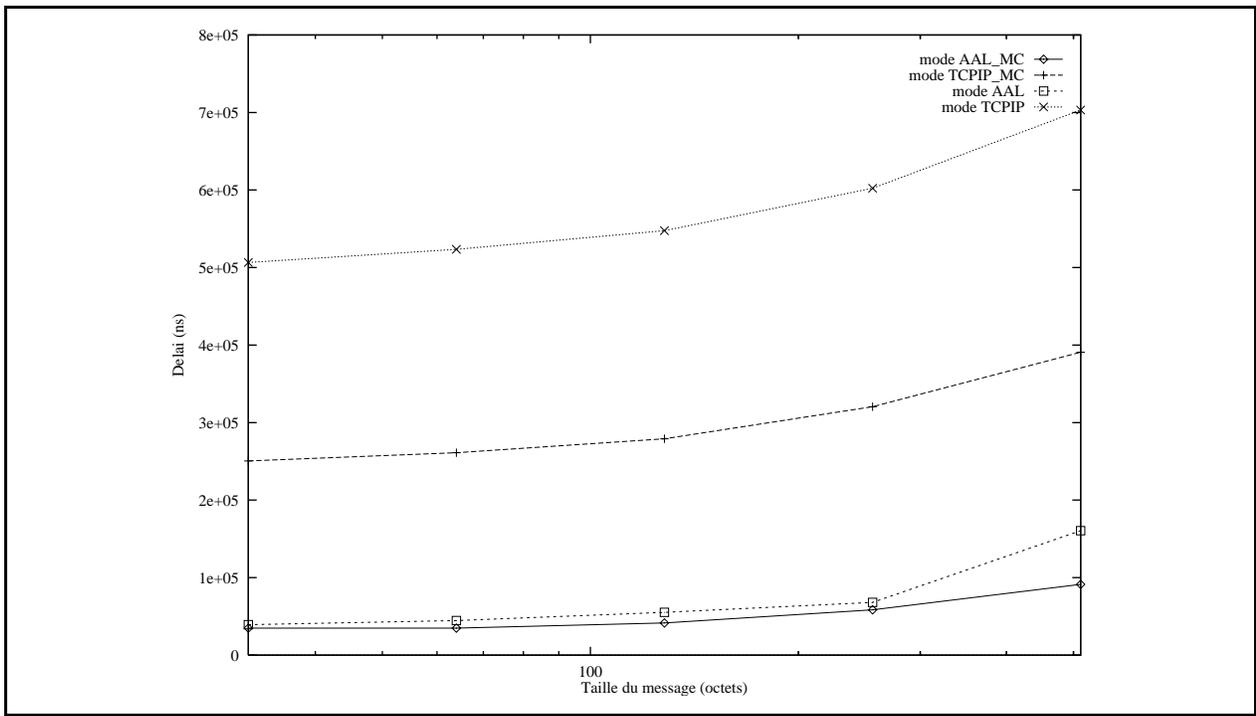


Figure IV.4b : Zoom sur la figure IV.4a

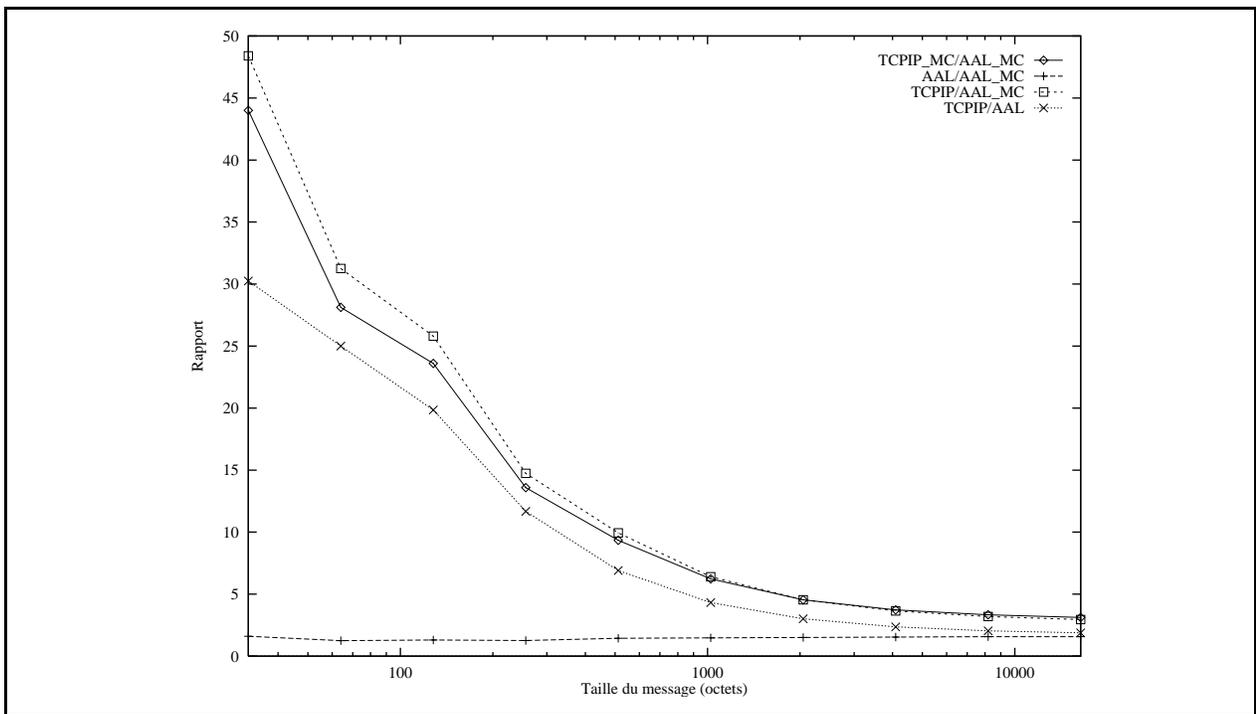


Figure IV.5 : Le rapport du débit maximum pour une connexion en émission

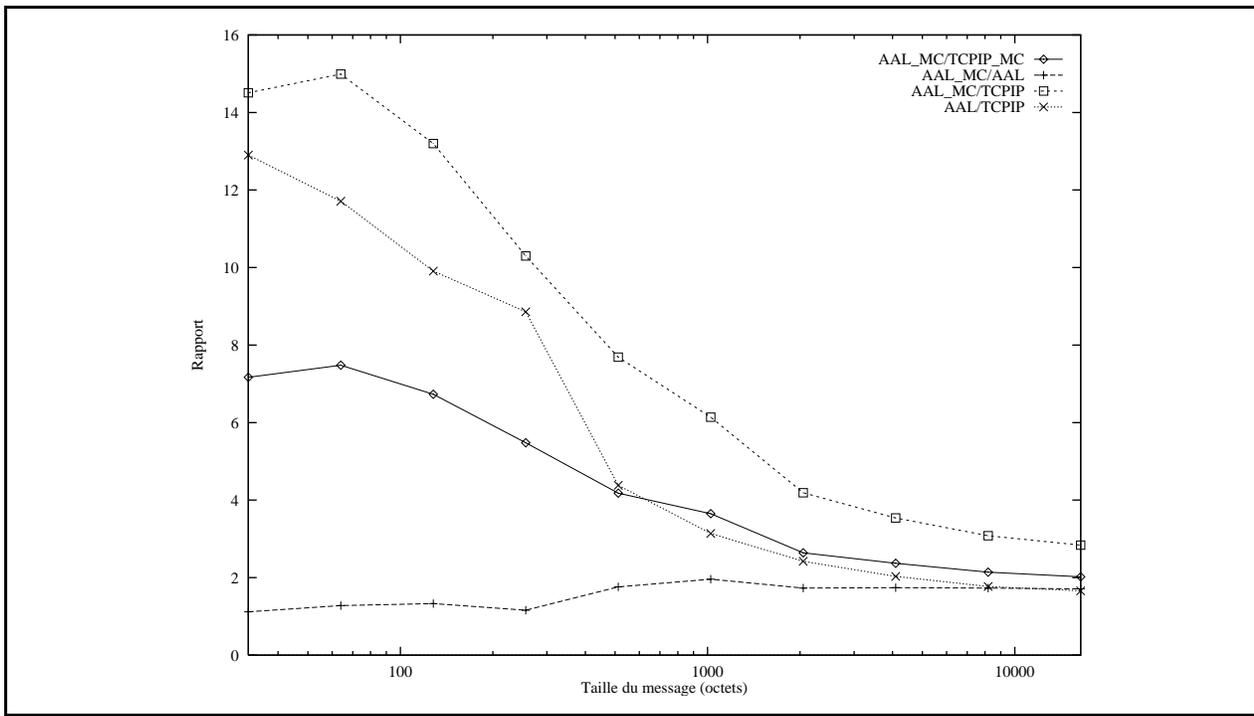


Figure IV.6 : Le rapport du délai de bout en bout pour une connexion en émission

Pour mieux comparer les différents modes de fonctionnement, il est aussi intéressant de connaître le rapport entre eux. Les figures IV.5 et IV.6 donnent respectivement le rapport pour le débit maximum et le rapport pour le délai de bout en bout.

Comme nous pouvions nous y attendre, le rapport entre les modes AAL et les modes TCP/IP décroît avec la taille des messages. Ceci est dû à l'amortissement du surcoût indépendant de la taille du message introduit par TCP/IP. D'autre part, ces courbes montrent que le fonctionnement avec le micro-contrôleur est deux à trois fois plus efficace que le fonctionnement dans l'autre configuration, lorsque le transfert est effectué directement sur l'AAL. Sur la courbe du gain du mode AAL_MC par rapport au mode AAL, les pertes d'efficacité dues à la taille des blocs transférés entre les deux tampons apparaissent nettement.

Néanmoins, le système de communication fonctionne rarement en émission seule. La figure IV.7 donne le débit maximum disponible dans le cas d'une connexion en émission et d'une connexion en réception sur chaque machine. Le débit maximum est alors de 90 Mbit/s pour les messages de grande taille, dans le mode AAL_MC. La saturation se produit au niveau du bus de la carte mère ATM. Lorsque le micro-contrôleur est absent, le débit maximum est inférieur de moitié au précédent. Dans ce cas, c'est le processeur hôte qui sature. La chute de débit pour le mode AAL pour les messages de 512

octets est due au traitement du vidage et du remplissage du tampon de segmentation et de réassemblage. En effet, ce transfert est réalisé par paquet de 10 cellules, soit 480 octets. Pour 512 octets, il faut un transfert de plus. Cette chute est moins sensible pour les multiples de 512 octets suivants puisqu'elle est amortie par le poids des autres traitements, en particulier la saturation du processeur hôte. Pour le mode AAL_MC, l'augmentation rapide du débit entre les messages de taille 1024 et 2048 octets est due à la longueur des blocs DMA entre le PC et la VRAM, égale à 2048 octets maximum. En effet, pour un message de 2048 octets, le coût de l'initialisation est mieux amorti.

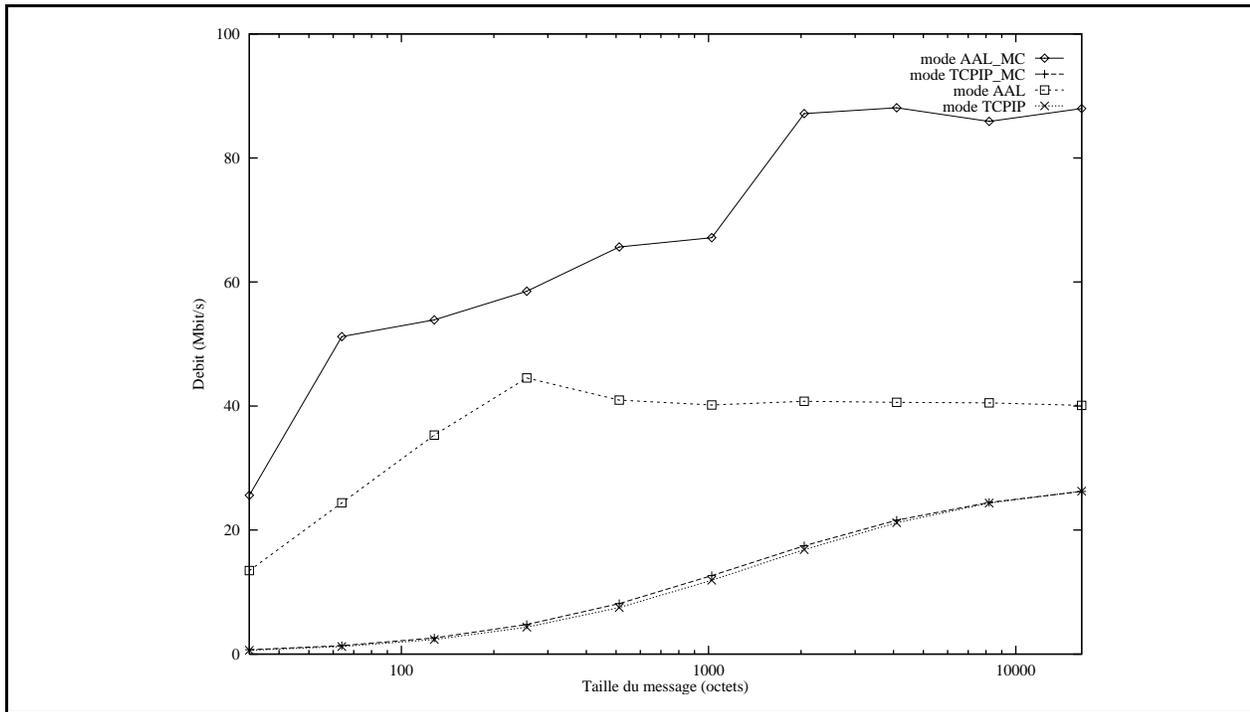


Figure IV.7 : Le débit maximum pour une connexion en émission et une en réception

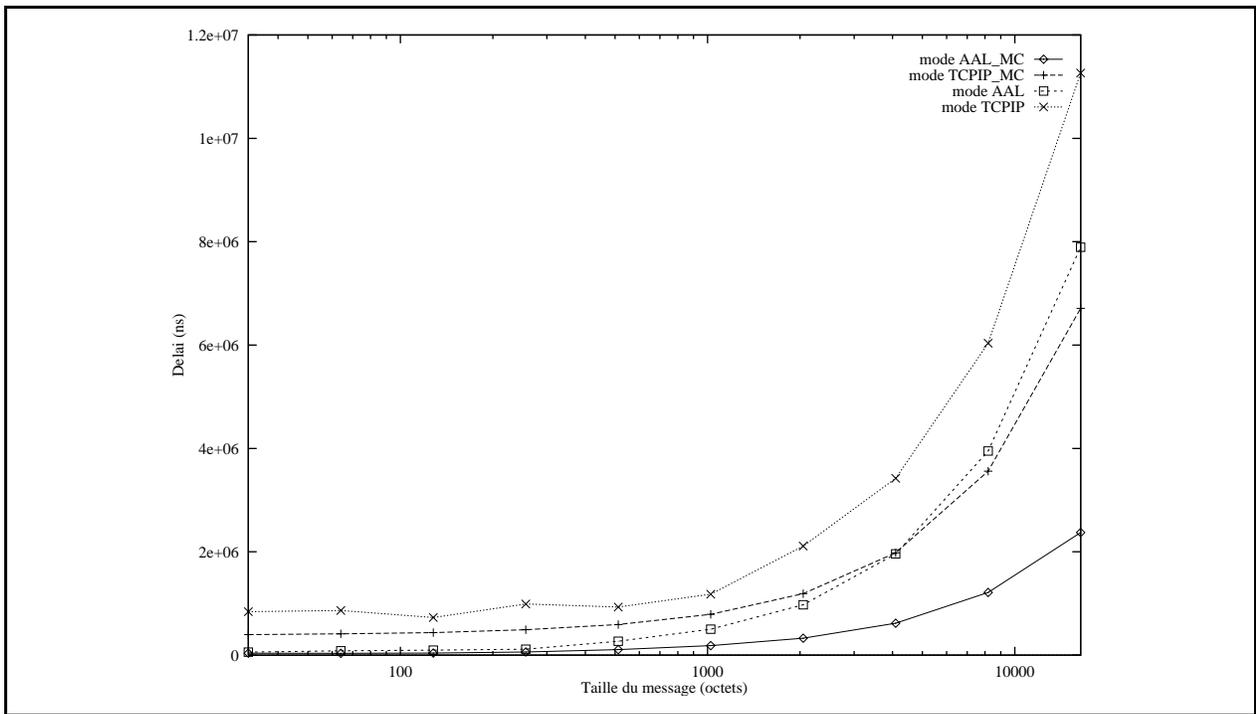


Figure IV.8 : Le délai de bout en bout pour une connexion en émission et une en réception

Dans cette configuration, le délai obtenu pour le mode AAL_MC est similaire à celui de l'émission seule (cf. figure IV.8). Par contre, celui de TCP/IP augmente de manière conséquente. Le rapport des délais ou des débits entre les différents modes est légèrement supérieur au cas avec une seule connexion en émission.

Jusqu'à présent, une seule connexion a été considérée. Regardons maintenant l'évolution des délais de transmission lorsque le nombre de connexions augmente. Les rapports de performances entre les modes AAL_MC et TCPIP_MC sont présentés sur la figure IV.9, pour 10, 25 et 50 connexions.

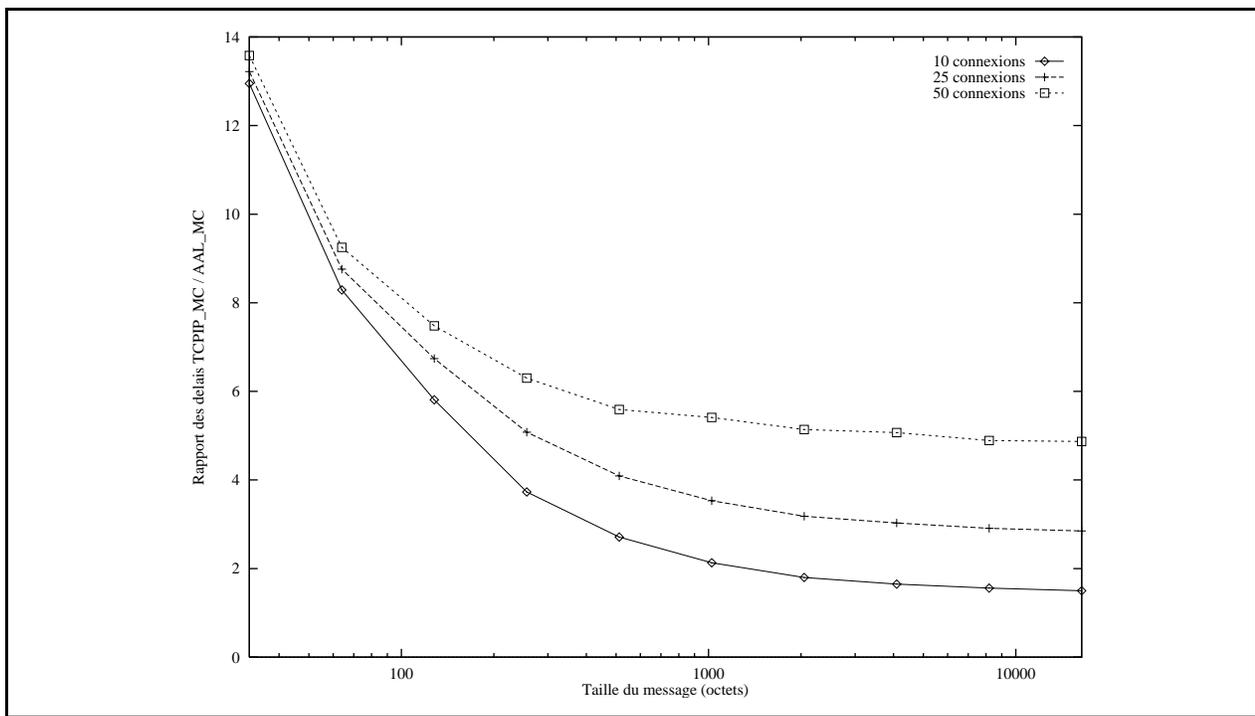


Figure IV.9 : L'évolution des délais en fonction du nombre de connexions

L'augmentation du nombre de connexions provoque une augmentation importante du délai de transmission de TCP/IP. Ainsi, le rapport entre les deux modes qui était de l'ordre de 2 pour de grands messages dans le cas d'une seule connexion devient supérieur à 5 pour 50 connexions. En effet, le protocole TCP/IP demande une capacité de traitement importante par connexion, indépendante de la longueur du message.

Nous avons vu dans ce paragraphe les caractéristiques de l'architecture en terme de délai de transmission et de débit, pour des messages classiques. Mais une des originalités du système de communication défini est le transfert critique d'informations. Le paragraphe suivant s'intéresse à ses performances.

IV.4.3. Les performances pour le transfert périodique de variables temps réel

Ce paragraphe compare les performances du transfert périodique de variables temps réel dans les quatre configurations. Le mode AAL_MC fonctionne selon la description du transfert critique effectuée dans le chapitre précédent. Les trois autres modes traitent les variables temps réel sur le processeur hôte, au niveau application.

Les deux paramètres importants à étudier sont le **cycle de rafraîchissement** et la **quantité de données**. Ce dernier est divisé en deux parties, le nombre de connexions

simultanées et le nombre de variables par connexion. Pour simplifier les comparaisons, chaque connexion transporte la même quantité de variables.

Ce paragraphe présente l'évolution d'un des paramètres en fixant les deux autres. Successivement sont examinées l'évolution de la période de rafraîchissement minimum, puis celle du nombre maximum de variables par connexion et enfin celle du nombre de connexions. Le paragraphe se termine par une comparaison des performances lorsque la charge et/ou la puissance du processeur hôte augmentent.

La table IV.1 présente l'évolution de la période du cycle de rafraîchissement en fonction du nombre de connexions et du nombre de variables par connexion. La quantité de variables est constante, fixée à 2000. Le processeur hôte exécute uniquement le protocole de communication et le transfert de variables. Aucune autre charge n'est présente.

Nombre de variables	400	200	100	50	20
Nombre de connexions	5	10	20	40	100
Période AAL_MC (ms)	0,57	0,52	0,51	0,55	0,55
Période TCPIP_MC (ms)	4,61	5,70	7,80	12,30	25,08
Période AAL (ms)	2,15	2,15	2,30	2,63	3,60
Période TCPIP (ms)	4,48	5,68	8,00	12,95	27,98
Gain TCPIP_MC/AAL_MC	8,09	10,96	15,29	22,36	45,60
Gain AAL/AAL_MC	3,77	4,13	4,51	4,78	7,20
Gain TCPIP/AAL_MC	7,85	10,92	15,68	23,54	50,87
Gain TCPIP/AAL	2,08	2,64	3,48	4,92	7,77

Table IV.1 : L'évolution de la période de rafraîchissement

Dans le cas du transfert critique, la période de rafraîchissement minimum est stable pour une quantité de variables donnée, quelle que soit la répartition entre le nombre de connexions et le nombre de variables pour chacune d'elles. Le traitement de ce service au plus près du réseau pour profiter de la gestion câblée des connexions ATM permet d'obtenir ces résultats. Lorsque le transfert passe par TCP/IP, la période minimum supportée augmente avec le nombre de connexions, les résultats étant similaires avec ou sans le micro-contrôleur local. Ceci est dû au protocole TCP/IP lui-même.

Lorsque le transfert périodique de variables est effectué dans la configuration sans micro-contrôleur, et directement sur l'AAL, la période minimum supportée est entre 4 et 7 fois plus grande que celle avec le transfert critique. Cette période augmente avec le

nombre de connexions. Dans ce cas, c'est le processeur hôte qui est saturé. Ces calculs ont été réalisés sans charge annexe sur le processeur.

Prenons le problème dans l'autre sens. Pour une période de rafraîchissement et un nombre de connexions fixés, cherchons le nombre de variables qui peuvent être transférées par connexion. La table IV.2 présente les résultats.

Nombre de connexions	20	20	50	100
Période (ms)	1	10	10	10
Nombre de variables AAL_MC	202	2052	862	430
Nombre de variables TCPIP_MC	impossible	141	impossible	impossible
Nombre de variables AAL	36	485	185	85
Nombre de variables TCPIP	impossible	136	impossible	impossible
Gain AAL_MC/TCPIP_MC	-	14,55	-	-
Gain AAL_MC/AAL	5,60	4,15	4,65	5,06
Gain AAL_MC/TCPIP	-	15,09	-	-
Gain AAL/TCPIP	-	3,56	-	-

Table IV.2 : L'évolution du nombre de variables par connexion

Pour une période de rafraîchissement de 1 ms, le protocole TCP/IP est incapable de suivre la fréquence imposée. Par contre le transfert critique supporte 20 connexions de 202 variables chacune, soit un peu plus de 4000 variables au total. Ce chiffre représente la quantité qui est généralement traitée par une station dans une application distribuée temps réel de ce type.

Lorsque la période est de 10 ms, TCP/IP supporte une vingtaine de connexions, mais est très vite limité lorsque ce nombre augmente. Les deux autres modes directement sur l'AAL permettent de transférer de 9000 (AAL) à 40000 (AAL_MC) variables avec une vingtaine de connexions. De plus, pour une période donnée, la quantité d'information (nombre de connexions x nombre de variables) transférée par le mode AAL_MC reste constante.

Jusqu'alors, la charge des processeurs n'a pas été considérée. Néanmoins, trois des quatre modes utilisent le processeur hôte de manière importante. De plus, les applications utilisant les variables temps réel s'exécutent sur le processeur hôte. Par conséquent, la table IV.3 donne le nombre de variables qui peuvent être transférées en fonction de la charge du processeur et du nombre de connexions.

Nombre de connexions	20	20	20	20
Charge du processeur hôte	20 %	20 %	50 %	50 %
Période (ms)	1	10	1	10
Nombre de variables AAL_MC	200	2028	125	1262
Nombre de variables TCPIP_MC	impossible	87	impossible	11
Nombre de variables AAL	26	380	7	240
Nombre de variables TCPIP	impossible	85	impossible	11
Gain AAL_MC/TCPIP_MC	-	23,31	-	114,72
Gain AAL_MC/AAL	7,69	5,33	17,86	5,26
Gain AAL_MC/TCPIP	-	23,86	-	114,72
Gain AAL/TCPIP	-	4,47	-	21,82

Table IV.3 : L'évolution du nombre de variables avec la charge du processeur

Lorsque le processeur est chargé, le transfert par TCP/IP devient très vite inefficace. Par contre, le transfert direct sur l'AAL sans micro-contrôleur reste stable pour des périodes de rafraîchissement de 10 ms. Il permet alors de transférer 5 fois moins de variables que le transfert critique. Par contre, pour des périodes plus petites, il arrive à saturation. L'écart avec le transfert critique est alors important.

Le transfert critique permet de transférer 4000 variables avec une période de rafraîchissement de 1 ms et une charge du processeur hôte de 20 %. Avec une charge de 50 %, cette valeur descend à 2500 variables.

Jusqu'alors, la puissance du processeur hôte était fixée à 50 MIPS. Devant l'évolution rapide des processeurs, il paraît intéressant d'étudier le comportement du système de transfert critique par rapport à des processeurs plus performants. Les tables IV.4 et IV.5 donnent respectivement les performances obtenues pour des processeurs hôtes de 100 MIPS et 200 MIPS.

Nombre de connexions	20	20	20	50	20	50
Charge du processeur hôte	0 %	0 %	20 %	20 %	50 %	50 %
Période (ms)	1	10	10	10	10	10
Nombre de variables AAL_MC	202	2052	2052	842	1323	527
Nombre de variables TCPIP_MC	impossible	355	263	40	124	impossible
Nombre de variables AAL	82	893	685	310	493	190
Nombre de variables TCPIP	impossible	460	330	55	170	impossible
Gain AAL_MC/TCPIP_MC	-	5,78	7,80	21,05	10,67	-
Gain AAL_MC/AAL	2,46	2,30	2,99	2,72	2,68	2,77
Gain AAL_MC/TCPIP	-	4,46	6,22	15,31	7,78	-
Gain AAL/TCPIP	-	1,94	2,08	5,64	2,90	-

Table IV.4 : L'évolution du nombre de variables pour un processeur hôte de 100 MIPS

Lorsque le processeur n'est pas chargé, le gain apporté par un processeur de 100 MIPS par rapport au processeur de 50 MIPS permet de transférer 3 fois plus de variables pour le traitement par TCP/IP. Pour un processeur chargé, le gain est encore plus important pour ces modes de traitement. Ceci se comprend bien puisque le protocole TCP/IP nécessite une capacité de traitement importante, dont une grande partie est indépendante de la quantité de données traitée.

Pour le mode AAL, le gain de performances est beaucoup plus faible, car le traitement relatif à ce mode effectué par le processeur hôte se limite à la gestion du tampon d'émission et de réception et à la communication avec la carte d'interface réseau. Néanmoins, les gains obtenus avec le micro-contrôleur par rapport à un traitement par TCP/IP sont encore importants puisqu'ils sont supérieurs à 10 lorsque le processeur hôte est chargé (50 %) ou lorsque le nombre de connexions simultanées est grand (50). De plus, un processeur de 100 MIPS n'est pas suffisant pour garantir une période de rafraîchissement de 1 ms avec le protocole TCP/IP.

Nombre de connexions	20	20	20	50	100
Charge du processeur hôte	20 %	50 %	50 %	50 %	50 %
Période (ms)	10	10	1	10	10
Nombre de variables AAL_MC	2052	1358	129	541	270
Nombre de variables TCPIP_MC	506	285	impossible	62	impossible
Nombre de variables AAL	815	627	50	275	120
Nombre de variables TCPIP	801	476	impossible	90	impossible
Gain AAL_MC/TCPIP_MC	4,06	4,77	-	8,73	-
Gain AAL_MC/AAL	2,52	2,17	2,58	1,97	2,25
Gain AAL_MC/TCPIP	2,56	2,85	-	6,01	-
Gain AAL/TCPIP	1,02	1,32	-	3,06	-

Table IV.5 : L'évolution du nombre de variables pour un processeur hôte de 200 MIPS

Lorsque l'on double encore la puissance du processeur hôte, les observations sont similaires à celles du passage d'un processeur de 50 MIPS à un processeur de 100 MIPS. L'écart entre les modes TCP/IP et les modes AAL se réduit. Néanmoins, dans cette situation, la quantité de données traitée, ou par analogie la taille des messages, est grande. Ce cas n'est pas favorable au micro-contrôleur comme l'ont montré les figures IV.5 et IV.6. Cependant, même dans ces conditions, lorsque le processeur hôte est chargé, le gain obtenu avec le micro-contrôleur est encore élevé (cf. table IV.5). De plus, avec le mode AAL_MC, le produit des trois termes : nombre de connexions, nombre de variables par connexion et fréquence de rafraîchissement (inverse de la période) reste constant. D'autre part, si la période de rafraîchissement est faible (1 ms), le micro-contrôleur est le seul à pouvoir effectuer les traitements nécessaires.

IV.5. Conclusion

Ce chapitre a présenté dans un premier temps le modèle du système de communication pour l'évaluation de ses performances. Les simulations ont permis de dresser les caractéristiques de l'architecture.

Ainsi, il est possible d'atteindre le débit maximal possible (140 Mbit/s) en émission au niveau de l'application. Lorsque l'émission et la réception sont combinées, le débit maximal supporté par l'architecture est de 90 Mbit/s.

L'évaluation des performances du service de transfert critique a montré que ce dernier était environ dix fois plus rapide que l'implantation classique avec IP sur ATM, même lorsque le processeur hôte est puissant (200 MIPS). D'autre part, en comparaison avec une implantation de ce service sur le processeur de la machine hôte, sans le micro-contrôleur, le service de transfert critique est 2 à 5 fois plus rapide pour de grandes quantités de données. Il est beaucoup plus efficace lorsque la quantité de variables par connexion est petite et le nombre de connexions grand. Pour des périodes de rafraîchissement très faible (1 ms), le gain apporté par le micro-contrôleur devient important. Surtout, le dispositif matériel est le seul à pouvoir transférer un nombre conséquent de variables, de 2000 à 4000 suivant la charge du processeur hôte, à des périodes de 1 ms. Enfin, le produit de la quantité globale de variables transférée par la fréquence de rafraîchissement reste constant avec le service de transfert critique, quel que soit le nombre de connexions simultanées.

Après avoir étudié les performances de l'architecture, il reste à comparer le coût du système à celui des systèmes existants. C'est l'objet du chapitre suivant.

Chapitre V

L'étude de coût

V.1. Introduction

Lorsque l'on crée une nouvelle architecture, il est toujours important de comparer le rapport performances / coût de celle-ci par rapport aux solutions existantes. Cependant, les fonctions des solutions sont parfois très éloignées et il est difficile de donner des valeurs comparables pour ce rapport.

Ce chapitre s'attache à la comparaison du coût de la solution proposée à celui de solutions disponibles sur le marché, en précisant les différences majeures entre les solutions comparées. Il s'intéresse plus particulièrement aux aspects réseau.

La solution type du marché est composée d'une carte d'interface au réseau ATM pour chaque machine et d'un ou plusieurs commutateurs. Pour faciliter les calculs, les éléments sont décomposés en trois parties :

- Le châssis du commutateur (pas de port ATM).
- Le port ATM pour le commutateur permettant d'interfacer une machine.
- La carte d'interface au réseau ATM.

La deuxième solution est celle proposée dans cette thèse et est constituée seulement de la carte de communication principale.

Les prix des équipements ATM variant très rapidement, il ne m'est pas paru utile de donner les valeurs en francs. Tous les coûts sont donc ramenés à une unité commune. On peut supposer que les différents constructeurs répercuteront la baisse des coûts dans les mêmes mesures. Néanmoins, cette étude a pour base les tarifs de juillet 1995.

Les différents coûts utiles aux calculs sont donnés dans la table V.1. Toutes ces valeurs correspondent au prix de vente des différents éléments, et non pas seulement à des coûts de développement. Pour les solutions du commerce, c'est le prix le plus bas disponible actuellement qui a été pris en compte.

Elément	Coût (en unité)
Châssis	90
Port ATM	5
Carte d'interface ATM	7
Carte sans micro-contrôleur	15
Carte avec micro-contrôleur	20

Table V.1 : Les coûts des éléments considérés

Tous les équipements utilisent ATM à **155 Mbit/s**. Le châssis du commutateur permet d'accueillir **16 ports** ATM maximum.

Les paragraphes suivants comparent les coûts dans deux situations : la système de communication sans le micro-contrôleur local, puis le système de communication avec le micro-contrôleur local.

V.2. Le coût de la solution sans le micro-contrôleur local

Lorsque l'on considère la solution proposée dans la configuration sans le micro-contrôleur local, celle-ci s'avère plus économique dès qu'il y a au moins trois machines (cf. figure V.1). En effet, pour deux machines, la solution classique ne nécessite pas de commutateur, mais ce n'est alors pas vraiment un réseau. Les sauts observés dans la courbe de la solution classique sont dus à l'ajout d'un châssis de commutateur lorsque tous les ports sont occupés. Ces sauts apparaissent pour un nombre de stations égal à 17, 31, 45, 59, 73, 87, ...

Plus le nombre de stations augmente, plus l'écart entre les deux solutions s'accroît. La différence atteint 390 unités pour 100 stations (1890 unités pour la solution classique et 1500 unités pour la solution proposée). Du point de vue des fonctions, les deux solutions sont à peu près comparables.

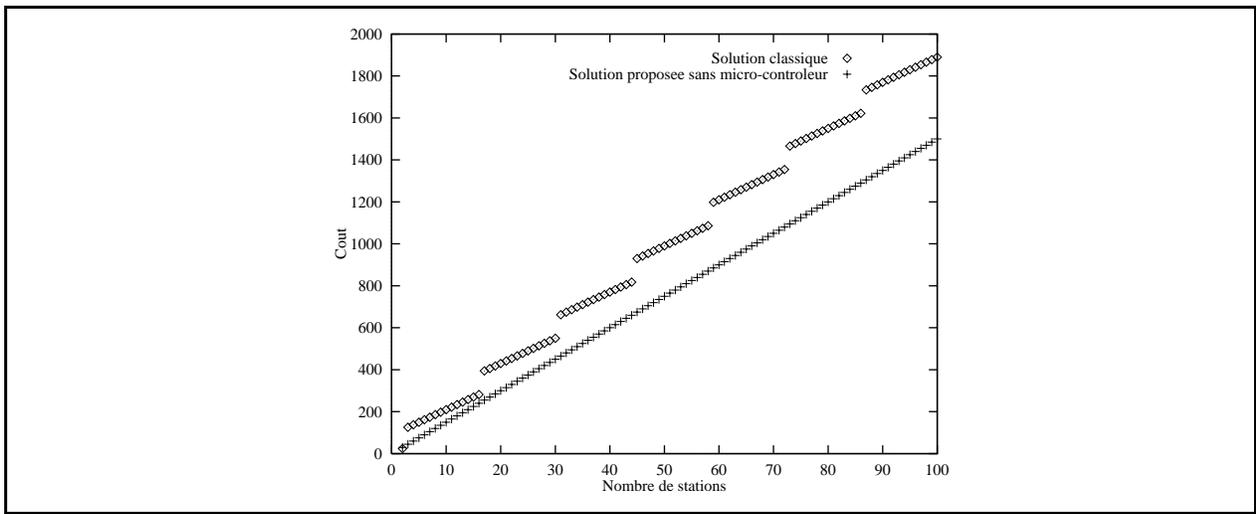


Figure V.1 : La comparaison des coûts sans le micro-contrôleur local

V.3. Le coût de la solution avec le micro-contrôleur local

Dans le cas de la solution avec micro-contrôleur local, les courbes des coûts sont proches en moyenne (cf. figure V.2). La solution proposée a tendance à augmenter un peu plus vite avec le nombre de stations. Cependant, pour 100 stations, la différence de coût est de 110 unités (2000 unités pour la solution proposée et 1890 unités pour la solution classique).

Toutefois, les fonctions des deux solutions sont cette fois très différentes. Le micro-contrôleur offre par exemple des services temps réel de haut niveau performants.

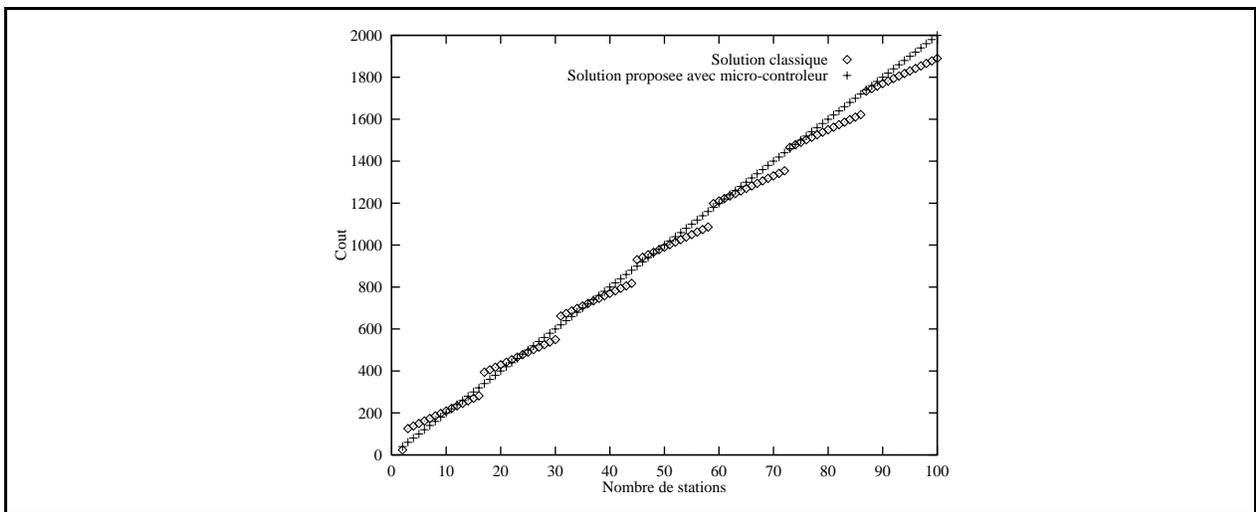


Figure V.2 : La comparaison des coûts avec le micro-contrôleur local

V.4. Conclusion

En conclusion de cette étude de coût, on peut dire que la solution proposée, à fonctionnalités égales, a un coût plus faible qu'une solution classique avec un commutateur externe. En effet, pour 100 stations, le coût est 26% plus élevé pour la solution classique.

Lorsque l'option du micro-contrôleur est ajoutée, la différence s'inverse au profit de la solution classique. Cependant, les fonctions des deux solutions sont cette fois très éloignées. De plus, pour 100 stations, le surcoût est limité à 5,8%.

Chapitre VI

Conclusion

Cette thèse a défini l'architecture d'un système de communication fondé sur ATM à 155,52 Mbit/s pour répondre aux nouveaux besoins des applications distribuées temps réel. L'architecture supporte les contraintes particulières du milieu industriel exprimées dans les objectifs initiaux :

- C1 Plusieurs équipements audio/vidéo sur une même station.
- C2 Topologies variées.
- C3 Redondance.
- C4 Qualités de service pour les applications temps réel.
- C5 Applications de téléinformatique classique.
- C6 Coût.

De nombreux projets ont proposé de nouvelles architectures de machines pour intégrer les flux multimédia. Ces architectures visent à faire passer les informations audio et vidéo par un chemin différent du bus fond de panier de la machine. Dans la plupart des cas, il s'agit d'un transfert direct vers le réseau. J'ai montré (cf. table II.2) qu'il n'existait pas une solution permettant de recouvrir l'ensemble des caractéristiques énumérées ci-avant. Par conséquent, j'ai été amené à définir une nouvelle architecture de système de communication.

VI.1. Les résultats

L'**architecture de couplage des équipements** comporte trois aspects importants : un *commutateur ATM embarqué*, un *micro-contrôleur local*, et une *organisation protocolaire* particulière.

Le *commutateur ATM* permet de disposer d'un *second bus* à l'intérieur de la machine. Ainsi, il est possible de supporter plusieurs équipements audio/vidéo sur une même station sans écrouler les performances de la machine (*caractéristique C1*). Les caméras peuvent alors être utilisées comme des capteurs d'images, similaires aux capteurs actuels, au débit transféré près.

De plus, il permet de construire des *topologies variées* (*caractéristique C2*) (en fait, toutes les sortes de topologies sont possibles), grâce aux accès multiples au réseau. L'absence de machines de commutation externes permet d'obtenir des structures linéaires (exemple de la télé-surveillance d'une autoroute ou d'un tunnel) ou des topologies quelconques. Le réseau peut-être ainsi composé uniquement de stations (il est possible, si on le souhaite, d'ajouter des commutateurs externes).

La structure proposée permet aussi de disposer de topologies à chemin d'accès différents, afin d'offrir la redondance (*caractéristique C3*).

D'autre part, la présence d'un *micro-contrôleur* sur la carte d'interface au réseau permet, en option, d'offrir des *services de haut niveau implantés au plus près du réseau*, grâce à une *organisation protocolaire* adaptée de la couche SSCS. Ces services sont basés sur les caractéristiques de la technologie ATM qui contient des fonctions des couches 3 et 4 du modèle OSI. Deux exemples de services ont été présentés. Ainsi, le transfert périodique de variables temps réel, ou les transferts audio/vidéo, ne subissent pas le traitement parfois lourd de toutes les couches de protocoles et du système d'exploitation. Seul un traitement minimal est effectué sur ces trafics. Le service se comporte alors comme un *télé-DMA*, c'est à dire que la fonction de DMA est étendue au réseau.

Aussi, l'implantation des protocoles critiques sur le micro-contrôleur permet de *garantir les contraintes temps réel (caractéristique C4)*. Les protocoles pour les données sans contrainte de délai s'exécutent sur le processeur hôte. Il est ainsi possible d'utiliser les protocoles intégrés au système d'exploitation (UNIX, Windows NT, ...). L'organisation protocolaire proposée permet de *supporter TCP/IP (caractéristique C5)*.

Un aspect important de l'architecture de couplage des équipements proposée est qu'elle peut-être supportée par n'importe quelle *machine standard*.

De plus, la structure proposée a un *coût plus faible* (à fonctionnalités égales, c'est à dire sans le micro-contrôleur) que celui des topologies de réseaux ATM classiques (*caractéristique C6*), puisque le commutateur externe n'est pas indispensable.

D'autre part, trois **mécanismes pour la gestion de la communication entre les équipements** ont été définis.

Un *mécanisme de contrôle de flux à la source* pour garantir le délai d'acheminement des messages temps réel a été proposé. Il permet d'éviter la perte de cellules due à des débordements de files au niveau des commutateurs. Il apporte une solution pour le support du trafic temps réel sur un réseau ATM. Cependant, il enlève un peu de souplesse au mécanisme d'allocation des débits offert par ATM (limitation de la longueur des rafales) et réduit le nombre de connexions, même si ce dernier reste assez élevé. Néanmoins, dans un système temps réel, il est préférable d'assurer que le message arrive dans un temps fini plutôt que d'optimiser l'utilisation de la bande passante.

De plus, un mécanisme pour augmenter la souplesse du contrôle d'accès au réseau a été présenté. Il consiste à autoriser des messages sans contrainte de temps à utiliser la bande passante disponible sur le réseau à un instant donné. Cependant, il n'y a alors

aucune garantie sur l'acheminement du message. En effet, si une cellule du message se trouve dans un commutateur subissant une congestion, cette cellule sera supprimée afin d'assurer que le trafic temps réel passe, et une retransmission du message devra être effectuée.

L'architecture originale du système de communication a conduit à la définition d'une signalisation particulière, puisque le commutateur embarqué impose à chaque station de disposer des protocoles UNI et NNI. Un *mécanisme d'ouverture de connexions par une tierce partie* a été ajouté afin de répondre aux besoins de configuration des équipements distants.

Enfin, un *mécanisme de gestion du basculement pour la redondance* des chemins a été défini. Il permet d'apporter plus de sûreté dans le fonctionnement du système de communication.

Ces trois mécanismes sont des solutions propriétaires qui ont été définies puisqu'aucune norme sur ces aspects n'est disponible. Si au cours du processus de normalisation d'ATM ces sujets étaient traités, il conviendrait de s'adapter. Cependant, pour le dernier mécanisme, il est peut probable qu'une norme soit créée à ce propos puisqu'il est très spécifique au domaine d'application considéré.

L'évaluation des performances du système de communication a montré que l'architecture avec le micro-contrôleur supporte le débit maximal possible en émission, soit 140 Mbit/s environ. Lorsque l'émission et la réception sont combinées, un débit de 90 Mbit/s est atteint, avec un processeur hôte d'une puissance de 50 MIPS.

Pour le service de transfert critique implanté sur le micro-contrôleur, le produit de la fréquence du cycle de contrôle-commande par la quantité de variables est multiplié par un facteur 10 par rapport aux implantations actuelles utilisant IP sur ATM. Ce facteur est réduit de moitié si la comparaison est faite avec une implantation de ce service sur le processeur hôte, directe sur l'AAL.

Mais, pour des fréquences de rafraîchissement de variables élevées (période de 1 ms), le service de transfert critique est le seul à pouvoir traiter un nombre de variables conséquent (2000 à 4000). De plus, avec le micro-contrôleur, et contrairement aux autres implantations considérées, la quantité globale de données traitée reste constante même si le nombre de connexions augmente.

Les **réalisations effectuées** peuvent se découper en cinq points : Une première version de produit a été développée par ITMI-APTOR, sur la base de l'architecture définie au cours de cette thèse. Elle offre les caractéristiques citées, exceptée la carte micro-contrôleur pour laquelle seul un connecteur de raccordement est prévu.

La carte micro-contrôleur est définie et réalisée en collaboration avec le Laboratoire d'Informatique de Brest - Télécom Bretagne. Ma contribution sur cette partie a porté sur la spécification de l'architecture matérielle, et les aspects logiciels (pilote de la carte sur la machine hôte, gestion des communications et des tampons au niveau du Transputer). Quelques milliers de lignes de code C sont développées.

Par contre, les trois mécanismes de contrôle de flux, d'ouverture de connexion par une tierce partie et de gestion du basculement sont restés à l'état de spécification. Le comportement et les performances du mécanisme de contrôle de flux pourront être étudiés par simulation VHDL, par exemple en ajoutant la description du réseau au modèle défini. Les deux autres mécanismes pourront être validés directement après implantation sur une plate-forme constituée des cartes et logiciels développés.

D'autre part, des modélisations VHDL fines des protocoles ATM ont été faites ainsi qu'une modélisation de plus haut niveau pour l'évaluation des performances de l'architecture du système de communication. Au total, 8000 lignes de code VHDL ont été écrites.

Enfin, j'ai participé à la mise en place d'une plate-forme de télé-enseignement. Cette plate-forme est composée de quatre postes clients PC sous Windows NT munis chacun d'une carte ATM développée par ITMI-APTOR, et d'une station de travail pour le serveur vidéo. Elle permet à des étudiants de consulter un cours, avec simultanément le texte du cours, les transparents, et des séquences audio/vidéo sélectionnées pour avoir des explications approfondies du professeur.

VI.2. Les perspectives

Dans les **points importants restant à étudier**, citons l'impact de la gestion répartie de la signalisation, imposée par la norme, sur le contrôle de la redondance. En effet, dans le cas d'une signalisation centralisée, il est facile d'avoir une vision globale de la structure du réseau. Lors de l'ouverture d'une connexion redondante, le calcul de chemins passants par des points distincts est possible. Par contre, si le processus de signalisation est entièrement réparti sur l'ensemble des commutateurs du réseau, il est difficile de pouvoir garantir cette indépendance des chemins. Des études supplémentaires sont à mener sur ce point.

D'autre part, les problèmes d'interconnexion au réseau public ATM n'ont pas été abordés. En particulier, l'impact du trafic en provenance de ce réseau sur le système de communication serait à étudier, pour autoriser les applications largement distribuées.

De plus, l'adaptation de la messagerie MMS directement au-dessus d'ATM a été évoquée, mais les idées présentées demandent un complément d'étude pour être implantées.

Finalement, du point de vue des **perspectives de cette étude**, la flexibilité offerte par le couple Transputer + FPGA permet d'envisager des extensions au système très variées. Ainsi, de nouveaux services (horloge répartie, synchronisation de processus, mémoire virtuelle largement distribuée, sous-ensemble de PVM ou MPI, ...) pourront être développés et évalués. Le système offre une base d'expérimentation précieuse qui sera utilisée pour déterminer les sous-ensembles de services applicatifs qu'il est intéressant d'implanter au plus proche du réseau.

Références

- [Ada93] J.F. Adam, H.H. Houh, D.L. Tennenhouse - *"Experience with the VuNet: A Network Architecture for Distributed Multimedia Systems"* - 18th IEEE Annual Conference of Local Computer Network - Minneapolis - Septembre 1993 - p 70-76
- [Ada94] J.F. Adam, H.H. Houh, M. Ismert, D.L. Tennenhouse - *"A Network Architecture for Distributed Multimedia Systems"* - MIT - 1994
- [Arm90] G.J. Armitage, K.M. Adams - *"Architecture of a Multimedia Desktop Workstation"* - Australian Video Communications Conference - Melbourne - Juillet 1990
- [Arm91] G.J. Armitage, K.M. Adams - *"Implementing an ATM Adaptation Layer in a Multi-media Terminal"* - Australian Broadband Switching and Services Symposium - Sidney - Juillet 1991
- [Arn89] E.A. Arnould, F.J. Bitz, E.C. Cooper, H.T. Kung, R.D. Sansom, P.A. Steenkiste - *"The Design of Nectar: A Network Backplane for Heterogeneous Multicomputers"* - Rapport de recherche CMU-CS-89-101 - Computer Science Department - CMU - Janvier 1989
- [Bar94] P. Barham, M. Hayter, D. McAuley, I. Pratt - *"Devices on the Desk Area Network"* - Rapport de recherche - Cambridge University & University of Twente - Mars 1994
- [Bel91] *"Report on B-ISDN protocols for providing SMDS and exchange access SMDS"* - Rapport de recherche - Bellcore - Septembre 1991
- [Bia93] E. Biagioni, E. Cooper, R. Sansom - *"Designing a Practical ATM LAN"* - IEEE Network - Mars 1993 - vol. 7 - n° 2 -p 32-39
- [Bie90] E.W. Biersack, D.C. Feldmeier - *"Transport protocol issues for ATM based networks"* - Rapport - EFOCLAN - Munich - 27-29 Juin 1990
- [Bla93] G. Blair, A. Campbell, G. Coulson, F. Garcia, D. Hutchinson, A. Scott, D. Shepherd - *"A network interface unit to support continuous media"* - IEEE Journal on Selected Areas in Communications - vol. 11 - n° 2 - Février 1993 - p 264-275
- [Bla94] G. Blair, G. Coulson, P. Robin, M. Papathamas - *"Supporting end-to-end quality of service in a micro-kernel environment"* - 7èmes entretiens Jacques Cartier - Colloque Informatique Communicante et Systèmes Distribués - Grenoble - France - 30 Novembre - 2 Décembre 1994 - p 13-17
- [Boe94] S. Boecking et al. - *"The BERKOM multimedia transport system"* - IS&T/SPIE Symposium on Electronic Imaging: Science & Technology - Février 1994 - p 256-265

- [Bra92] T. Bradely, C. Brown - "*Inverse Address Resolution Protocol*" - RFC 1293 - USC/Information Sciences Institute - Janvier 1992
- [Bri91] M. Brill, U. Gramm - "*MMS : MAP application services for the manufacturing industry*" - Computer Networks and ISDN Systems - vol. 21 - n° 5 - Juillet 1991 - p 357-380
- [Bry92] J. Bryan Lyles, D.C. Swinehart - "*The emerging gigabit environment and the role of local ATM*" - IEEE Communications Magazine - Avril 1992 - p 52-58
- [Brz92] T. Braun, M. Zitterbart - "*Parallel transport system design*" - Proc. of 4th IFIP conference on High Performance Networking - Liège (Belgique) - 14-18 Décembre 1992 - p H3.1-H3.16
- [Cam92] A. Campbell, G. Coulson, F. Garcia, D. Hutchinson - "*A Continuous Media Transport and Orchestration Service*" - ACM SIGCOMM'92 - Baltimore - Maryland - Août 1992
- [Cher89] D. Cheriton, C. Williamson - "*VMTP as the transport layer for high performance Distributed Systems*" - IEEE Communications magazine - vol. 27 - n° 6 - Juin 1989
- [Cla87] D.D. Clark, M. Lambert, L. Zhang - "*NETBLT: A Bulk Transfer Protocol*" - RFC 998 - Mars 1987
- [Cla89] D.C. Clark, V. Jacobson, J. romkey, H. Salwen - "*An Analysis of TCP Processing Overhead*" - IEEE Communications Magazine - vol. 27 - n° 6 - Juin 1989 - p 23-29
- [Cla90] D.D. Clark, D.L. Tennenhouse - "*Architectural Considerations for a New Generation of Protocols*" - Proceedings of ACM SIGCOMM'90 - Septembre 1990 - p 200-208
- [Cla94] P. Claridge, D. Holloway - "*LAN Emulation Tutorial*" - Contribution ATM-Forum 94-0225 - 10 Mars 1994
- [Coh92] D. Cohen, G. Finn, R. Felderman, A. DeSchon - "*ATOMIC : a local communication network created through repeated application of multicomputing components*" - Rapport de recherche - University of Southern California - Janvier 1992
- [Coh93] D. Cohen, G. Finn, R. Felderman, A. DeSchon - "*ATOMIC: a low cost, very high speed Local Communication Architecture*" - International Conference on Parallel Processing - 1993
- [Col95] R.G. Cole, D. Shur - "*IP Over ATM: A Framework Document*" - Internet Draft - version 5 - AT&T Bell Laboratories - 5 Septembre 1995
- [Coo90] E.C. Cooper et al. - "*Protocol Implementation on the Nectar Communication Processor*" - SIGCOMM'90 - 1990 - p 135-143

- [Dab92] W. Dabbous, C. Huitema, L. Villader Siso, J. Seoane, J. Berrocal - *"Applicability of the session and the presentation layers for the support of high speed applications"* - Rapport de recherche n° 144 - INRIA Sophia Antipolis - Octobre 1992
- [Dav91] B.S. Davie - *"A Host-Network Interface Architecture for ATM"* - Proceedings of ACM SIGCOMM'91 - Zurich - 3-6 Septembre 1991 - p 307-315
- [Dav94] B. Davie - *"Host Interfacing for High Speed Networks"* - Tutorial - 5th International IFIP Conference on High Performance Networking - Grenoble (France) - 27 Juin - 1 Juillet 1994
- [Del92] L. Delgrossi, R.G. Herrwitch, F.O. Hoffmann - *"An implementation of ST-II for the Heidelberg transport system"* - GLOBECOM'92 - Orlando - 1992
- [Dos92] B.T. Doshi, P.K. Johri - *"Communication protocols for high speed packet networks"* - Computer Networks and ISDN Systems - vol. 24 - n° 3 - Mai 1992 - p 243-273
- [Ets92] ETSI DE/NA-53203 - *"Network Aspects (NA) - CBDS over ATM"* - draft - Décembre 1992
- [Fel93] D.C. Feldmeier - *"An Overview of the TP++ Transport Protocol Project"* - Computer Communication Research Group - Bellcore - 1993
- [Fin91] G. Finn, - *"An integration of network communication with workstation architecture"* - Rapport de recherche - University of Southern California - Mars 1991 --- ACM Computer Communication Review - vol. 21 - n° 5 - Octobre 1991
- [For94] *"200-Series ATM Adapter: Design and Architecture"* - Fore Systems - 1994
- [Fua93] *"Adaptation Layer Controller (ALC)"* - Data sheet - Fujitsu - 1993
- [Fub93] *"ATM Switch Element (SRE)"* - Data sheet - Fujitsu - 1993
- [Fuc93] *"Address Translation Controller (ATC)"* - Data sheet - Fujitsu - 1993
- [Fud93] *"Network Termination Controller (NTC)"* - Data sheet - Fujitsu - 1993
- [Gal91] D. Le Gall - *"MPEG: A video compression standard for multimedia applications"* - Communications of ACM - vol. 34 - n° 4 - Avril 1991 - p 46-58
- [Gmc92] D.J. Greaves, D. McAuley - *"Private ATM networks"* - IFIP 3rd International Workshop on Protocols for High Speed Networks - Stockholm - Mai 1992 --- University of Cambridge - Computer Laboratory - Technical Report n° 258 - Mai 1992
- [Gre92] D.J. Greaves, D. McAuley, L.J. French - *"Protocol and interface for ATM LANs"* - 5th IEEE Workshop on Metropolitan Area Networks - Taormina - Italy - Mai 1992 --- University of Cambridge - Computer Laboratory - Technical Report n° 258 - Mai 1992

- [Haa91] Z. Haas - *"A Protocol Structure for High-Speed Communication over Broadband ISDN"* - IEEE Networks - vol. 5 - n° 1 - Janvier 1991 - p 64-70
- [Hay91] M. Hayter, D. McAuley - *"The Desk Area Network"* - ACM Operating Systems Review - vol. 25 - n° 4 - Octobre 1991 - p 14-21
- [Hay93] M.D. Hayter - *"A Workstation Architecture to Support Multimedia"* - Thèse TR-319 - University of Cambridge - Septembre 1993
- [Heh90] D. Hehmann, M. Salmony, H. Stuttgen - *"Transport Services for Multimedia Applications on Broadband Networks"* - Computer Communications - vol. 13 - n° 4 - Mai 1990
- [Hei94] J. Heinanen - *"NBMA address resolution protocol (NBMA ARP)"* - Internet Draft - Version 1 - Telecom Finland - 8 Février 1994
- [Hop88] A. Hopper, R.M. Needham - *"The Cambridge Fast Ring Network"* - IEEE Transactions on Computers - vol. 37 - n° 10 - Octobre 1988 - p 1214-1223
- [Hop90] A. Hopper - *"Pandora - an experimental system for multimedia applications"* - ACM Operating Systems Review - vol. 24 - n° 2 - Avril 1990 - p 19-34
- [I.363] Recommandation UIT-T I.363 - *Spécification de la couche d'adaptation ATM du RNIS Large Bande* - Mars 1993
- [I.610] Recommandation UIT-T I.610 - *"Principes d'exploitation et de maintenance pour l'accès au RNIS Large Bande"* - Mars 1993
- [IISP95] ATM-Forum - *"Interim Inter-Switch Signaling Protocol"* - Février 1995
- [ISO11172] ISO 11172 - *Codage des images numériques animées et du son associé pour les supports de stockage numérique, jusqu'à environ 1,5 Mbit/s*
- [ISO8649] ISO 8649 - Service definition for the Association Control Service Element
- [ISO8650] ISO 8650 - Protocol specification for the Association Control Service Element
- [ISO9506-1] ISO 9506 - Industrial Automation Systems - Manufacturing Message Specification - Part 1: Protocol Definition - 22 Décembre 1988
- [ISO9506-2] ISO 9506 - Industrial Automation Systems - Manufacturing Message Specification - Part 2: Service Definition - 22 Décembre 1988
- [Jac90] V. Jacobson - *"4 BSD TCP Header Prediction"* - Computer Communication Review - vol. 20 - n° 2 - Avril 1990 - p 13-15
- [Jeo94] B.C. Jeon, D.Y. Kim - *"HMR: An ATM-Based Medium Access Protocol for Gigabit Ring Networks"* - 5th International IFIP Conference on High Performance Networking - Grenoble - France - 27 Juin - 1 Juillet 1994 - p 1-14

- [Jon93] A. Jones, A. Hopper - *"Handling Audio and Video Streams in a Distributed Environment"* - ACM SIGOPS - 1993 - p 231-243
- [Kal91] B.S. Kaliski - *"A Layman's Guide to a subset of ASN.1, BER, and DER"* - 3 Juin 1991
- [Kan88] H. Kanakia, D.R. Cheriton - *"The VMP Network Adapter Board (NAB): High Performance Network Communication for Multiprocessors"* - SIGCOMM'88 - Stanford - Californie - 16-19 Août 1988 - p 175-187
- [Kan91] H.R. Kanakia - *"High Performance Host Interfacing for Packet-Switched Networks"* - Thèse - STAN-CS-91-1373 - Stanford University - Juillet 1991
- [Kat94] D. Katz, D. Piscitello - *"NBMA Next Hop Resolution Protocol (NHRP)"* - Internet Draft - Novembre 1994
- [Law94] A. Lawrence, A. Kay, W. Luk, T. Nomura, I. Page - *"Using Reconfigurable Hardware to Speed up Product Development and Performance"* - JFIT conference - Edinburgh - Mars 1994
- [Lem93] M. Lemercier, G. Pujolle - *"Contrôle d'accès dans les réseaux ATM, études des performances des "leaky buckets" "* - Réseaux et informatique répartie - vol. 3 - n° 3 - 1993 - p 267-286
- [Lem95] ATM-Forum - *"LAN Emulation over ATM Specification - version 1"* - Février 1995
- [Lep89] F. Lepage, F. Afilal, P. Antoine, E. Bajic, J.Y. Bron, T. Divoux - *"Les réseaux locaux industriels"* - Ed. Hermes - 1989
- [Les91] I. Leslie, D. McAuley - *"Fairisle : an ATM network for the local area"* - Proceedings of ACM SIGCOMM'91 - Zurich - 3-6 Septembre 1991 - p 327-336
- [Les92] I.M. Leslie, D. McAuley, S.J. Mullender - *"Pegasus - Operating System Support for Distributed Multimedia Systems"* - Rapport de recherche - Cambridge University & University of Twente - 16 Novembre 1992
- [Lia93] F. Liaw - *"IP over ATM: architecture, address translation and call control"* - Internet Draft - Version 0 - Sun Microsystems - 21 Mars 1993
- [Lid94] C.J. Lindblad, D. Wetherall, and D.L. Tennenhouse - *"The VuSystem: A Programming System for Visual Processing of Digital Video"* - ACM Multimedia 94 - San Francisco - Californie - Octobre 1994
- [Lie92] S. Liew, K.M. Adams - *"Hardware design aspects in implementing ATM functions in a multimedia workstation"* - papier soumis à 2nd Australian Multimedia Communications, Applications and Technology Workshop 1992

- [Lie93] S. Liew, K.M. Adams - *"Completion of a prototype ATM cell-stream splitter for an experimental multimedia terminal"* - papier soumis à 3rd Australian Multimedia Communications, Applications and Technology Workshop 1993
- [Lin94] C.J. Lindblad, D. Wetherall, W. Stasior, B. Phillips, D. Bacher, J.F. Adam, H.H. Houh, M. Ismert, D.L. Tennenhouse - *"ViewStation Applications: Intelligent Video Processing Over A Broadband Local Area Network"* - USENIX Symposium on High-Speed Networking - Oakland - Californie - 1-3 Août 1994
- [Lin95] M. Lin, J. Hsieh, D.H.C Du, J.P. Thomas, J.A. McDonald - *"Distributed Network Computing over Local ATM Networks"* - IEEE Journal on Selected Areas in Communications - 1995
- [Mca89] D.R. McAuley - *"Protocol Design for High Speed Networks"* - Thèse TR-186 - University of Cambridge - Septembre 1989
- [Mca93] D. McAuley - *"Operating System Support for the Desk Area Network"* - 4th International Workshop on Network and OS Support for Digital Audio and Video - Lancaster - 3-5 Novembre 1993 - p 13-20
- [Mic93] J.R. Michel - *"The Design and Evaluation of an Off-Host Communications Protocol Architecture"* - Master of Science - University of Virginia - Août 1993
- [Min94] R. Minnich, D. Burns, F. Hady - *"A 1.2 GBit/sec, 1 microsecond latency ATM interface"* - USENIX Symposium on High-Speed Networking - Oakland - Californie - 1-3 Août 1994 - p 73-89
- [Mit94] D.J. Mitzel, D. Estrin, S. Shenker, L. Zhang - *"An Architectural Comparison of ST-II and RSVP"* - Soumis à INFOCOM 1994
- [Mpe93] ISO/IEC JTC1/SC29/WG11/N0531 - *MPEG-2 Systems* - Septembre 1993
- [Mur87] K.H. Muralidhar - *"MAP 2.1 network management and directory services test system"* - Proc. of Protocol Specification, Testing and Verification - 1987 - p 359-369
- [Oka91] T. Okada, H. Ohnishi, N. Morita - *"Traffic Control in Asynchronous Transfer Mode"* - IEEE Communications Magazine - Septembre 1991 - vol. 29 - n° 9 - p 58-62
- [Par93] C. Partridge - *"Protocols for high-speed networks: some questions and a few answers"* - Computer Networks and ISDN Systems - vol. 24 - Avril 1993
- [PCI93] *PCI Local Bus Specification* - Revision 2.0 - 30 Avril 1993
- [Pfe93] T. Pfeifer - *"Micronet Machines - New Architectural Approaches for Multimedia End-systems"* - 4th International Workshop on Network and OS Support for Digital Audio and Video - Lancaster - 3-5 Novembre 1993 - p 29-40

- [Plu82] D. Plummer - *"An Ethernet Address Resolution Protocol - or - Converting Network Addresses to 48 bit Ethernet Address for Transmission on Ethernet Hardware"* - RFC 826 - MIT - Novembre 1992
- [Pot93] B. Pottier, J.M. Filloque - *"ArMen : une machine parallèle et polymorphe"* - Journées du PRC-GdR Architectures Nouvelles de Machines - Rennes - 6-7 Décembre 1993
- [Prt93] C. Partridge - *"Gigabit Networking"* - Addison-Wesley - Octobre 1993
- [Pry91] M. De Prycker - *"Asynchronous Transfer Mode : solution for broadband ISDN"* - Ellis Horwood Editor - 1991
- [Puj94] G. Pujolle - *"Les réseaux large bande et l'ATM"* - X^o congrès De Nouvelles Architectures pour les Communications - Paris - 17-19 Mai 1994 - p 1-37
- [Q.2100] ITU-T Recommendation Q.2100 - *B-ISDN Signaling ATM Adaptation Layer Overview Description* - Décembre 1993
- [Q.2110] ITU-T Recommendation Q.2110 - *B-ISDN - ATM Adaptation Layer - Service Specific Connection Oriented Protocol (SSCOP)* - Décembre 1993
- [Q.2130] ITU-T Recommendation Q.2130 - *B-ISDN Signalling ATM Adaptation Layer - Service Specific Coordination Function for Support of Signalling at the User-to-Network Interface (SSCF. at UNI)* - Décembre 1993
- [Q.2140] ITU-T Recommendation Q.2140 - *B-ISDN Signalling ATM Adaptation Layer - Service Specific Coordination Function for Support of Signalling at the Network Node Interface (SSCF. at NNI)*
- [Q.2931] ITU-T Recommendation Q.2931 - *Broadband Integrated Services Digital Network (B-ISDN) Digital Subscriber Signalling System No. 2 (DSS 2) User Network Interface Layer 3 Specification for Basic Call/Connection Control* - Draft - Juin 1994
- [Q.93b] ITU-T Recommendation Q.93b - *"B-ISDN User Network Interface Layer 3 Specification for basic call/bearer control"* - Draft Octobre 1992
- [Ram93] K.K. Ramakrishnam - *"Performance considerations in designing network interfaces"* - IEEE Journal on Selected Areas in Communications - vol. 11 - n^o 2 - Février 1993 - p 203-219
- [RFC1483] J. Heinanen - *"Multiprotocol encapsulation over ATM Adaptation Layer 5"* - RFC 1483 - Telecom Finland - Juillet 1993
- [RFC1577] M. Laubach - *"Classical IP and ARP over ATM"* - RFC 1577 - Hewlett Packard - Janvier 1994
- [RFC1595] T. Brown, K. Tesink - *"Definitions of Managed Objects for the SONET/SDH Interface Type"* - RFC 1595 - 11 Mars 1994

- [RFC1626] R.J. Atkinson - *"Default IP MTU for use over ATM AAL5"* - RFC 1626 - Naval Research Laboratory - Mai 1994
- [RFC1695] M. Ahmed, K. Tesink - *"Definitions of Managed Objects for ATM Management Version 8.0 using SMIv2"* - RFC 1695 - 25 Août 1994
- [RFC1755] M. Perez, F. Liaw, A. Mankin, E. Hoffman, D. Grossman, A. Malis - *"ATM Signalling Support for IP over ATM"* - RFC 1755 - Février 1995
- [Ric93] W.D. Richard, J.R. Cox, B.L. Gottlieb, K. Krieger - *"The Washington University Multimedia System"* - Multimedia Systems - 1993 - p 120-131
- [Rob94] P. Robin, G. Coulson, A. Campbell, G. Blair, M. Papathomas, D. Hutchinson - *"Implementing a QoS Controlled ATM Based Communications System in Chorus"* - 4th International IFIP Workshop on Protocols for High Speed Networks (PfHSN) - Vancouver - Canada - 10-12 Août 1994 - p 19-35
- [Rol88] P. Rolin - *"Réseaux locaux : normes et protocoles"* - Ed. Hermes - 1988
- [Ros86] F.E. Ross - *"FDDI : a tutorial"* - IEEE Computer Magazine - n° 24 - 1986
- [Sab93] H. Sabourin - *"Anneau à compensation"* - CFIP'93 - Montréal - Canada - 7-9 Septembre 1993
- [Sac94] M.W. Sachs, A. Leff, D. Sevigny - *"LAN and I/O Convergence: A Survey of the Issues"* - IEEE Computer - Décembre 1994 - p 24-32
- [Sak93] S. Sakata - *"B-ISDN Multimedia Workstation Architecture"* - IEEE Communications Magazine - Août 1993 - p 64-67
- [Ste94] P.A. Steenkiste - *"A Systematic Approach to Host Interface Design for High-Speed Networks"* - IEEE Computer - Mars 1994 - p 47-57
- [Sto93] Stodolsky et al. - *"Fast Interrupt Priority Management in Operating System Kernels"* - Usenix Symposium on Microkernels and Other Kernel Architecture - San Diego - California - 20-21 Septembre 1993 - p 105-110
- [Tan90] A. Tanenbaum - *"Réseaux : architectures, protocoles, applications"* - InterEditions - 1990
- [Tan93] A. Tantawy - *"Réalisation de protocoles à haute performance"* - CFIP'93 - Montréal - 7-9 Septembre 1993 - p 115-126
- [Taw92] W. Tawbi, S. Dupuy, E. Horlait - *"High speed protocols: state of the art in multimedia applications"* - Proc. of 4th Information Network and Data Communication - Espoo - Finlande - 16-19 Mars 1992 - p 91-106
- [Tho93] J.P. Thomesse - *"le réseau de terrain FIP"* - Réseaux et informatique répartie - vol. 3 - n° 3 - 1993 - p 287-321

- [Tra92] C.B.S. Traw - *"A Host Interface Architecture and Implementation for ATM Networks"* - Rapport technique - University of Pennsylvania - Mai 1992
- [Tra94] J.M. Tracey, A. Banerji - *"Device Driver Issues in High-Performance Networking"* - USENIX Symposium on High-Speed Networking - Oakland - Californie - 1-3 Août 1994
- [Tub91] A. Tubtiang, H.I. Kwon, G. Pujolle - *"A method for classification of ATM switches"* - Rapport de recherche - Institut Blaise Pascal - Paris - Décembre 1991
- [Tur86] J. Turner - *"New directions in communications (or which way in the information age?)"* - IEEE Communications Magazine - vol. 24 - Octobre 1986 - p 8-15
- [Tsw92] TranSwitch Corporation - *"SARA Chipset Technical Manual"* - 2 Novembre 1992
- [UNI3.1] ATM-Forum - *"ATM User-Network Interface specification - version 3.1"* - Janvier 1995
- [Vog94] A. Vogel, G.V. Bochmann, J. Gecsei, B. Kerherve - *"Distributed Multimedia Applications and Quality of Service - A Survey -"* - 7èmes entretiens Jacques Cartier - Colloque Informatique Communicante et Systèmes Distribués - Grenoble - France - 30 Novembre - 2 Décembre 1994 - p 45-58
- [Wol93] A. Wolman, G. Voelker, C.A. Thekkath - *"Latency Analysis of TCP on an ATM Network"* - Rapport UW-CSE-93-03-03 - Dept. of Computer Science and Engineering - University of Washington - Mars 1993
- [XTP92] *XTP Protocol Definition - Revision 3.6 - 1992*
- [802.5] IEEE 802.5 - *"Token Ring Access Method"* - 1985

Annexe A

Glossaire des abréviations utilisées

AAL	ATM Adaptation Layer	Couche d'adaptation à ATM
ABR	Available Bit Rate	Débit disponible
ACSE	Association Control Service Element	Elément du service de contrôle d'association
ANSI	American National Standards Institute	Organisme américain de standardisation
API	Application Program Interface	Interface de programmation
ARP	Address Resolution Protocol	Protocole de résolution d'adresses
ASN	Abstract Syntax Notation	Syntaxe abstraite
ATM	Asynchronous Transfer Mode	Mode de transfert asynchrone
B-ISDN	Broadband aspects of Integrated Services Digital Network	Aspects large bande du réseau numérique à intégration de service large bande
BUS	Broadcast and Unknown Server	Serveur de diffusion du LAN Emulation
CAC	Connection Admission Control	Contrôle d'admission des connexions
CBDS	Connectionless Broadband Data Service	Service de données sans connexion large bande
CBR	Constant Bit Rate	Débit constant
CCITT		Comité Consultatif International Télégraphique et Téléphonique
CIM	Computer Integrated Manufacturing	Informatique de production intégrée
CLNAP	Connectionless Network Access Protocol	Protocole sans connexion
CLP	Cell Loss Priority	Priorité à la perte de cellules
CLS	Connectionless Server	Serveur sans connexion
CMIP	Common Management Information Protocol	Protocole d'administration commun
CPCS	Common Part Convergence Sublayer	Partie commune de la sous-couche de convergence
CRC	Cyclic Redundancy Code	Code de redondance cyclique

CS	Convergence Sublayer	Sous-couche de convergence
DAN	Desk Area Network	Réseau de bureau
DMA	Direct Memory Access	Accès direct mémoire
DPRAM	Dual Port Random Access Memory	Mémoire double accès
DQDB	Distributed Queue Dual Bus	Bus à double anneau
DSP	Data signal Processor	Processeur de traitement du signal
EISA	Enhanced ISA	Bus ISA amélioré
ETSI	European Telecommunications Standards Institute	Organisme européen de standardisation pour les télécommunications
FDDI	Fibre Distributed Data Interface	Interface de données à distribution de fibres
FIFO	First In First Out	File d'attente
FIP	Factory Instrumentation Protocol	Flux d'informations en provenance ou vers le processus
FPGA	Field Programmable Gate Array	Composant programmable
FRM	Fast Resource Management	Gestion rapide des ressources
GFC	Generic Flow Control	Champ de contrôle de flux
HEC	Header Error Check	Code de détection d'erreur sur l'en- tête
HIPPI	High Performance Parallel Interface	Interface parallèle haute performance
IETF	Internet Engineering Task Force	Groupe de travail Internet
IP	Internet Protocol	Protocole Internet
ISO	International Standard Organization	Organisation de standardisation internationale
ITU-T	International Telecommunication Union - Telecommunication Section	Union Internationale des Télécommunications - Secteur des télécommunications
LAN	Local Area Network	Réseau local
LEC	LAN Emulation Client	Client pour le LAN Emulation
LECS	LAN Emulation Configuration Server	Serveur de Configuration pour le LAN Emulation

LES	LAN Emulation Server	Serveur pour le LAN Emulation
LIS	Logical IP Subnetwork	Sous-réseau IP logique
LLC	Logical Link Control	Sous-couche de contrôle de liaison
MAC	Medium Access Control	Contrôle d'accès au médium
MAN	Metropolitan Area Network	Réseau de zone urbaine
MAP	Manufacturing Automation Protocol	Protocole pour les automatismes industriels
MIB	Management Information Base	Base d'informations d'administration
MID	Multiplexing Identification	Identificateur de multiplexage
MIS	Management Information Server	Serveur d'informations d'administration
MMS	Manufacturing Message Specification	Spécification de messagerie industrielle
MPEG	Moving Picture Expert Group	Groupe d'experts pour les images animées
MPI	Message Passing Interface	Interface pour le passage de messages
MSNA	Multi-Service Network Architecture	Architecture réseau multi-services
MTU	Maximum Transmission Unit	Unité de transmission maximum
NAB	Network Adapter Board	Carte d'adaptation réseau
NBMA	Non-Broadcast Multi-Access	Réseau sans diffusion, multi-accès
NDIS	Network Driver Interface Specification	Spécification d'une interface pour les pilotes réseau par Microsoft et 3COM
NETBLT	NETwork BLock Transfer	Transfert réseau par blocs
NHRP	Next Hop Resolution Protocol	Protocole de résolution sur le noeud suivant
NHS	NHRP Server	Serveur HPRP
NNI	Network-to-Network Interface	Interface de signalisation réseau
NPC	Network Parameter Control	Contrôle des paramètres de réseau
NRM	Network Resource Management	Gestion des ressources du réseau

NSAP	Network Service Access Point	Point d'accès au service réseau
OAM	Operation and Maintenance	Opération et maintenance
ODI	Open Data Link Interface	Spécification pour les pilotes réseau introduite par Novell
OSI	Open Systems Interconnection	Interconnexion des systèmes ouverts
PCI	Peripheral Component Interconnect	Bus d'interconnexion
PDH	Plesiochronous Digital Hierarchy	Hiérarchie digitale plesiochrone
PDU	Protocol Data Unit	Unité de données de protocole
PM	Physical Medium	Sous-couche support physique
POH	Path OverHead	Information sur un chemin SDH
PT	Payload Type	Type du contenu de la cellule
PVC	Permanent Virtual Connection	Connexion virtuelle permanente
PVM	Parallel Virtual Machine	Machine parallèle virtuelle
QOS	Quality of service	Qualité de service
RNIS-LB		Réseau Numérique à Intégration de Service Large Bande
RPC	Remote Procedure Call	Appel de procédures à distance
SAP	Service Access Point	Point d'accès de service
SAR	Segmentation And Reassembly	Segmentation et réassemblage
SCSI	Small Computer System Interface	Bus pour les périphériques
SDH	Synchronous Digital Hierarchy	Hiérarchie digitale synchrone
SDU	Service Data Unit	Unité de données de service
SIMM	Single Memory Module	Module mémoire
SMDS	Switched Multimegabit Data Service	Service commuté de données multi-mégabit
SNMP	Simple Network Management Protocol	Protocole d'administration de réseau simplifié
SOH	Section OverHead	Information sur une section SDH
SONET	Synchronous Optical NETWORK	Réseau optique synchrone
SSCF	Service Specific Coordination Function	Fonction de coordination spécifique au service

SSCOP	Service Specific Connection Oriented Protocol	Protocole connecté spécifique au service
SSCS	Specific Service Convergence Sublayer	Sous-couche de convergence spécifique au service
STM	Synchronous Transport Module	Module de transport pour SDH
STS	Synchronous Transport Signal	Signal de transport pour SONET
SVC	Switched Virtual Connection	Connexion virtuelle commutée pour IP sur ATM
SVC	Signalling Virtual Channel	Canal de communication pour la signalisation
TC	Transmission Convergence	Sous-couche de convergence de transmission
TCP	Transmission Control Protocol	Protocole de contrôle de transmission
UBR	Unspecified Bit Rate	Débit non spécifié
UDP	User Datagram Protocol	Protocole de transfert de datagrammes
UIT-T	International Telecommunication Union - Telecommunication Section	Union Internationale des Télécommunications - Secteur des télécommunications
UNI	User Network Interface	Interface utilisateur au réseau
UPC	Usage Parameter Control	Contrôle des paramètres d'utilisation
UTP	Unshielded Twisted Pair	Paire torsadée non blindée
VBR	Variable Bit Rate	Débit variable
VC	Virtual Channel	Voie virtuelle
VCC	Virtual Channel Connection	Connexion de voie virtuelle
VCI	Virtual Channel Identifier	Identificateur de voie virtuelle
VHDL	VHSIC Hardware Description Language	Langage de description d'architectures
VHSIC	Very High Speed Integration Circuits	Circuits intégrés rapides
VLSI	Very Large Scale Integration	Intégration à large échelle

VMTP	Versatile Message Transaction Protocol	Protocole pour la transaction de messages variés
VP	Virtual Channel	Conduit virtuel
VPC	Virtual Channel Connection	Connexion de conduit virtuel
VPI	Virtual Channel Identifier	Identificateur de conduit virtuel
VRAM	Vidéo Random Access Memory	Mémoire vidéo
WAN	Wide Area Network	Réseau étendu
XTP	Xpress Transfer Protocol	Protocole de transport

Annexe B

La technique ATM

B.1. Les caractéristiques d'ATM

B.1.1. Introduction

L'évolution des techniques de transmission, de codage et de compression des images et du son, et des supports de communication permettent aujourd'hui d'envisager de nouveaux réseaux de communication. Ainsi, la fibre optique permet d'augmenter considérablement les débits disponibles. Le codage et la compression des images permet une utilisation à large échelle de ce type d'information. Enfin, les techniques de transfert asynchrone, et ATM (Asynchronous Transfer Mode) en particulier, apporte une réponse au problème du transport d'informations de natures différentes.

Le futur Réseau Numérique à Intégration de Services Large Bande (RNIS-LB ou B-ISDN: Broadband Integrated Services Digital Network en anglais) va permettre de supporter un large éventail de services audio, vidéo et de transfert de données [Kun92]. Le mode de transfert asynchrone (ATM) a été choisi par le CCITT (aujourd'hui renommé UIT-T) pour ce réseau. ATM, qui a vu le jour au début des années 80 [Cne91], est un mode de transfert très souple qui n'impose pas un débit fixe des informations. Ceci est adéquat pour supporter tous types de services, aussi bien à débit constant que variable. ATM a été défini afin de répondre à un besoin en matière de services à haut débit. Son ambition est d'être multi-services, afin de pouvoir répondre à toute évolution future des demandes de service de la part des utilisateurs, le marché étant imprévisible.

On distingue généralement deux catégories de services susceptibles d'utiliser les réseaux de télécommunication [I.211] :

- Les services interactifs qui sont séparés en trois groupes :
 - Les services de conversation tels la visiophonie ou la visioconférence.
 - Les services de messagerie, par exemple les services postaux de transfert d'images haute résolution ou de films, les messageries vocales.
 - Les services de consultation, tels la consultation de films, d'images à haute résolution, de sons ou d'archives.
- Les services de distribution, qui regroupent :
 - Les services *sans commande* dans lesquels l'utilisateur peut accéder au service sans pouvoir déterminer à quel moment commencera la diffusion. C'est typiquement la radio ou la télévision classique.
 - Les services *avec commandes* dans lesquels l'information est diffusée de manière cyclique. L'utilisateur peut alors accéder individuellement à

l'information et commander le déclenchement. C'est le cas de la vidéographie diffusée sur canal complet.

Du côté des applications informatiques, non seulement les applications réparties passeront d'une répartition au niveau des réseaux locaux à une répartition au niveau national voire international. Mais, en plus, les technologies haut débit du RNIS-LB peuvent être adoptées dans les nouveaux réseaux locaux où l'architecture répartie des systèmes et des applications sera omniprésente.

Il faut donc prévoir une indépendance totale au niveau du protocole entre les propriétés de débit et la possibilité de diversité de services qu'il doit offrir. ATM a donc comme soucis majeur de s'adapter aux plus hauts débits accessibles à la technologie et de servir de support à des applications pouvant présenter de fortes contraintes temporelles. Tous types de données sont traités, aussi bien isochrones (son et image) qu'asynchrones (données) [Bou92]. ATM offre une grande variété de débits pour les utilisateurs, allant de quelques kilobits par seconde à quelques centaines de mégabits par seconde (155 et 622 Mbit/s voire 2,5 Gbit/s dans le futur pour les débits normalisés).

B.1.2. Les principes fondamentaux d'ATM

Le protocole de transfert choisi est le mode paquet, seul à pouvoir résoudre les problèmes d'incertitude et de variété des débits demandés par les utilisateurs auxquels on peut s'attendre. Il fait appel aux techniques de multiplexage asynchrone par répartition de temps, par opposition au multiplexage temporel synchrone et à la commutation de circuits. Il transfère l'information après l'avoir découpée en paquets dont la structure est fixe et courte (53 octets) afin de satisfaire au mieux les besoins en débit et simplifier les traitements associés. Ce paquet se nomme **cellule**. Sa taille fixe et courte permet d'augmenter la capacité de traitement des nœuds de commutation (parallélisme), et de réduire le temps de mise en paquet.

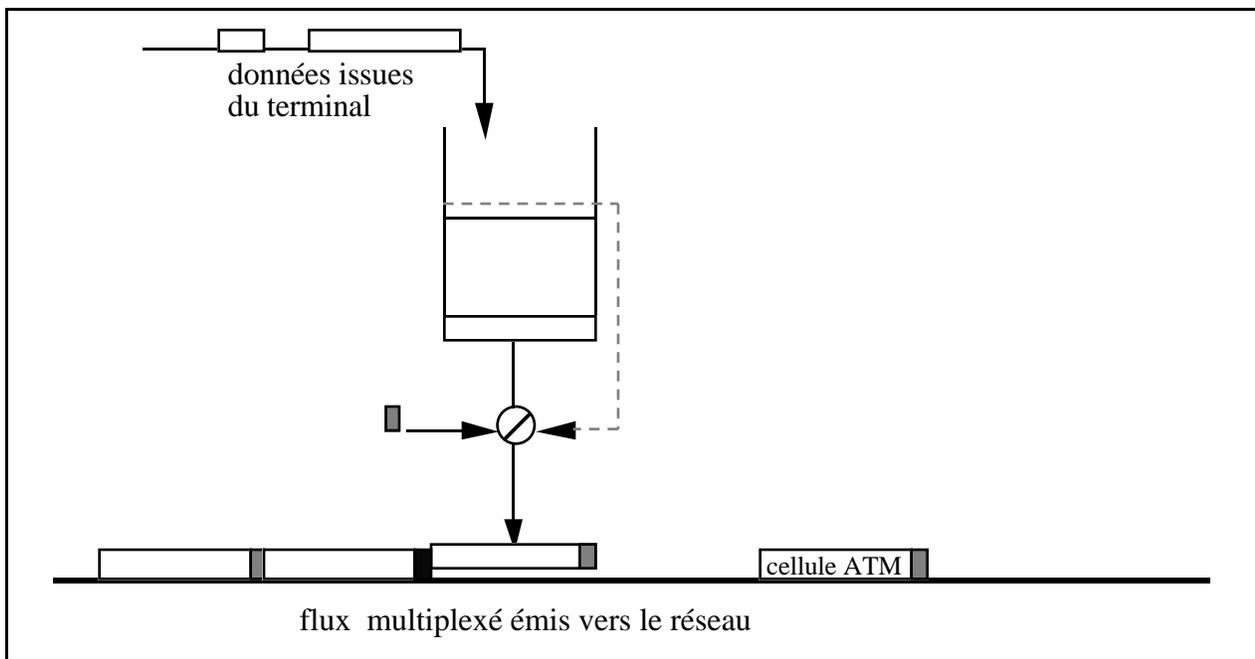


Figure B.1 : Le multiplexage des cellules ATM

Le format de la cellule ainsi que la prépondérance du temps de propagation physique demandé ont conduit à retenir une solution d'acheminement par **voie logique**. Pour cela, un identificateur spécifique de l'en-tête de la cellule ATM portant le numéro de voie logique (identifié par un couple chemin virtuel / voie virtuelle) est utilisé [I.150]. Cela implique que la communication ATM est à la base en **mode connecté**. Chaque cellule est alors acheminée individuellement sur sa voie logique, et l'on peut multiplexer divers flux d'informations sur un même support (cf. figure B.1).

Afin d'en faire un mode multi-services, simple et universel, on découple totalement :

- Les opérations d'acheminement.
- Les protocoles et fonctions dépendant des applications.

Pour cela, on découpe la cellule en deux champs :

- Un en-tête lié aux fonctions d'acheminement.
- Un champ d'information lié aux applications.

B.1.3. Le réseau ATM

Le réseau ATM est constitué d'un ensemble de **commutateurs**, formant un maillage physique quelconque, et reliés par des **liens séries mono-directionnels** (cf. figure B.2).

ATM étant un protocole en mode connecté, un identificateur de connexion contenu dans chaque en-tête de cellule permet d'acheminer les informations sur le réseau. Cet identificateur associe explicitement une cellule à une voie virtuelle sur un lien physique (cf. figure B.3). L'identificateur de la connexion est composé de deux parties : le VPI (Virtual Path Identifier) et le VCI (Virtual Channel Identifier) [Sch90]. Ces deux informations permettent de multiplexer, de démultiplexer et de diriger la cellule à travers le réseau. Cependant, les champs VPI et VCI ne sont pas des adresses, ce sont des étiquettes. Ils sont assignés à une connexion sur chaque segment entre deux nœuds du réseau lors de l'ouverture de la connexion. Les VP sont utilisés pour regrouper les VC. Bien que les cellules soient brassées tout au long de leur parcours sur le réseau, l'ordre de celles-ci sur un chemin virtuel donné n'est pas modifié entre la source et la destination.

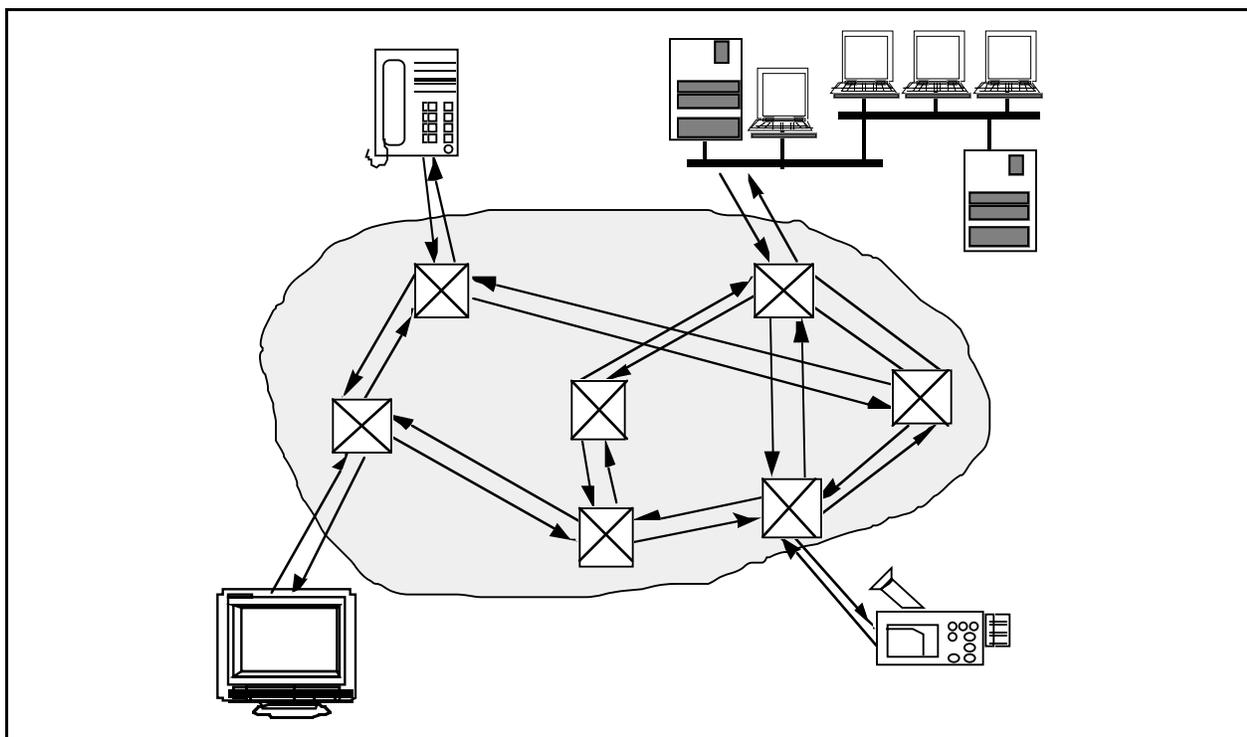


Figure B.2 : Schéma d'un réseau ATM

Le besoin multi-services du RNIS-LB amènera à prévoir aussi un mode non connecté, que l'on construira au-dessus du réseau public ATM [Bel91] (cf. paragraphe B.1.8).

B.1.4. Le modèle de référence du protocole ATM

Le modèle de référence du protocole ATM (cf. figure B.4) [I.321] a été établi suivant les principes de communication en couches de la Recommandation X.200 du CCITT (Modèle de référence pour l'interconnexion des systèmes ouverts (OSI) pour les applications) [Pry91]. La définition du modèle est en cours d'évolution, en particulier pour les couches hautes du protocole. Le modèle de référence se compose de trois plans :

- Le **plan usager**, avec sa structure en couches, assure le transfert du flux d'informations de l'utilisateur.
- Le **plan contrôle**, avec sa structure en couches, accomplit les fonctions de signalisation nécessaires à l'établissement, à la supervision et à la libération des connexions et des communications.
- Le **plan de gestion** exécute deux types de fonctions : les fonctions de gestion liées à l'ensemble du système, qui assurent la coordination entre tous les plans (Gestion de plan) et les fonctions de gestion liées à la maintenance et à l'exploitation de chaque couche (Gestion de couche).

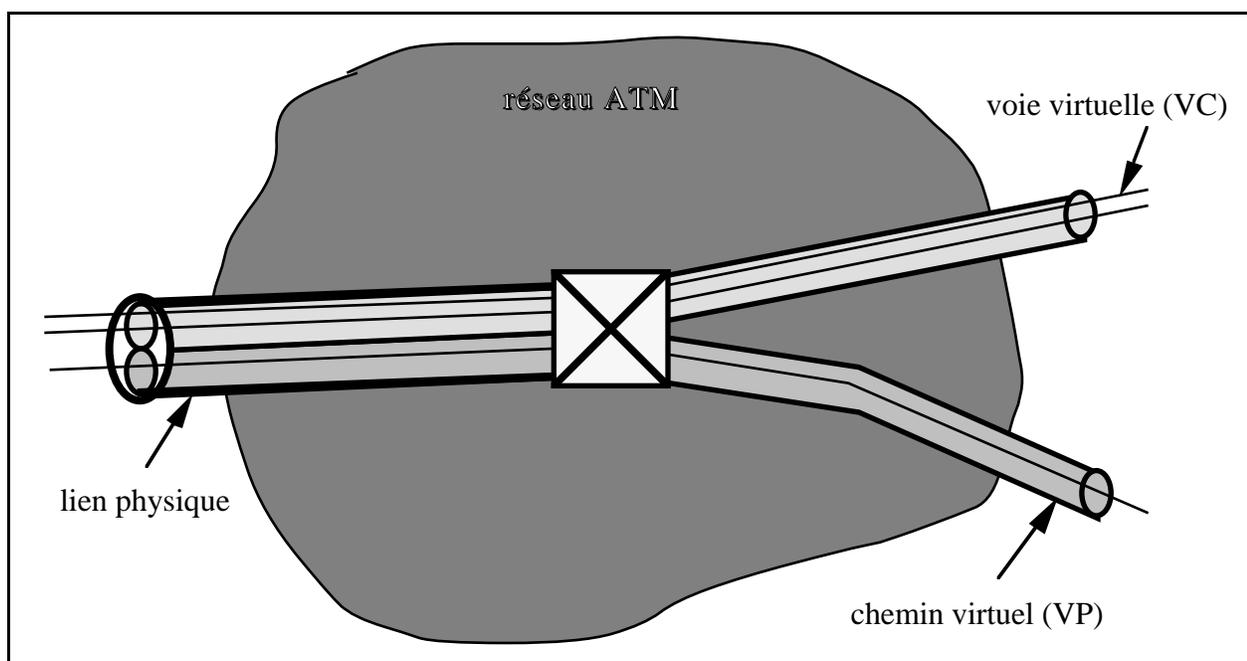


Figure B.3 : Le principe des voies virtuelles

Le modèle de référence du protocole comporte trois couches :

- La **couche physique** (PHY) réalise les fonctions liées à la transmission des éléments binaires sur un support physique.
- La **couche ATM**, commune à tous les services, effectue les traitements relatifs à la cellule.
- La **couche d'adaptation à ATM** (AAL) permet de supporter les différents types de services.

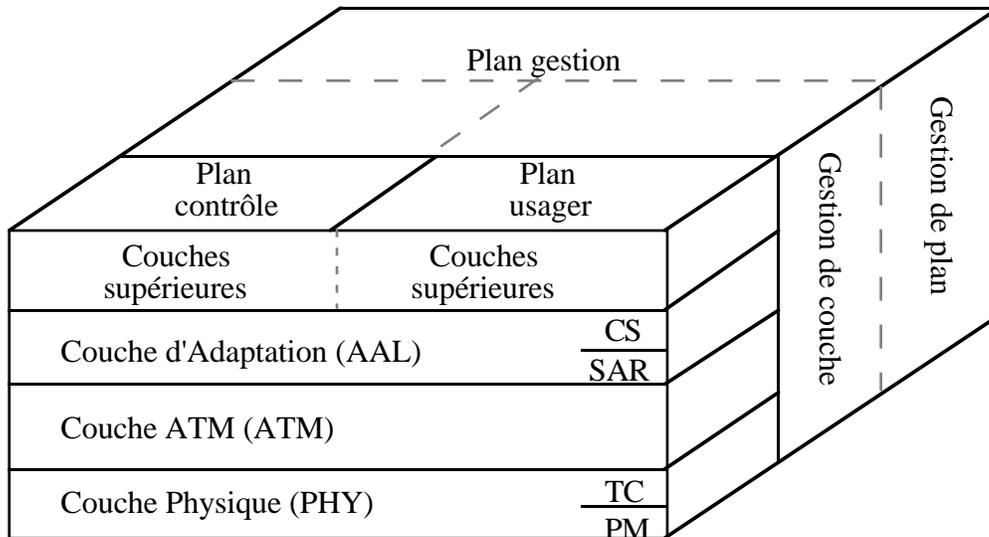


Figure B.4 : Le modèle de référence du protocole ATM

La nécessité de ce nouveau modèle provient du besoin de transférer les flux multimédia, c'est à dire une superposition de sons, d'images et de données. A l'opposé, le modèle OSI était pensé uniquement pour les transferts de données [Pry91]. De plus, le modèle de protocole ATM ne s'intéresse qu'au transport de bout en bout de l'information.

B.1.5. La couche physique

La couche Physique [I.432] permet l'émission et la réception des cellules ATM. Elle est composée de deux sous-couches :

- La sous-couche support physique (PM : Physical Medium) réalise la transmission des bits sur le réseau en fonction du médium utilisé.
- La sous-couche convergence de transmission (TC : Transmission Convergence) transfère le flot de bits disponible au niveau de l'interface avec la sous-couche PM. Elle le transforme en un flot de cellules valides qui seront utilisées par la couche ATM. Les fonctions réalisées par cette sous-couche incluent la génération de

cellules, la récupération et l'adaptation des données, la délimitation des cellules en provenance du réseau, la vérification et la génération du code d'erreur (HEC) pour les cellules et la gestion du débit des cellules (insertion de cellules vides lorsque c'est nécessaire).

La couche physique accepte de nombreux systèmes de transmission tels que la hiérarchie digitale synchrone (SDH), SONET ou le transfert basé sur des cellules.

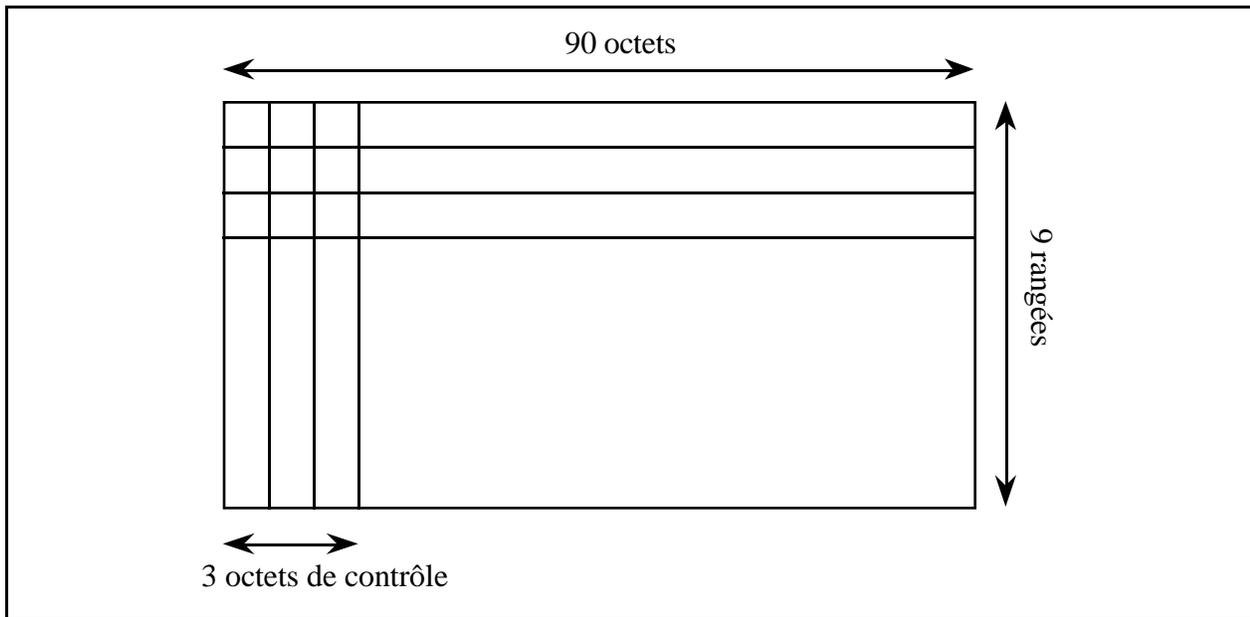


Figure B.5 : La trame SONET STS-1

SONET décrit la structure d'une trame synchrone qui est émise toutes les 125 μ s. La longueur de cette trame dépend de la vitesse de l'interface. La norme SONET a été difficile à établir puisque la hiérarchie des débits dans les trois continents (Europe, Etats-Unis, Japon) est très différente. Finalement, un débit de 51,84 Mbit/s a été choisi et forme le premier niveau (STS-1 : Synchronous Transport Signal 1). Une trame SONET STS-1 est composée de 9 rangées de 90 octets, 3 octets par rangée permettant le contrôle (cf. figure B.5). Les trames de niveaux supérieurs, notées STS-N, sont construites à partir de la trame STS-1 (cf. figure B.6). Les débits normalisés sont présentés dans la table B.1.

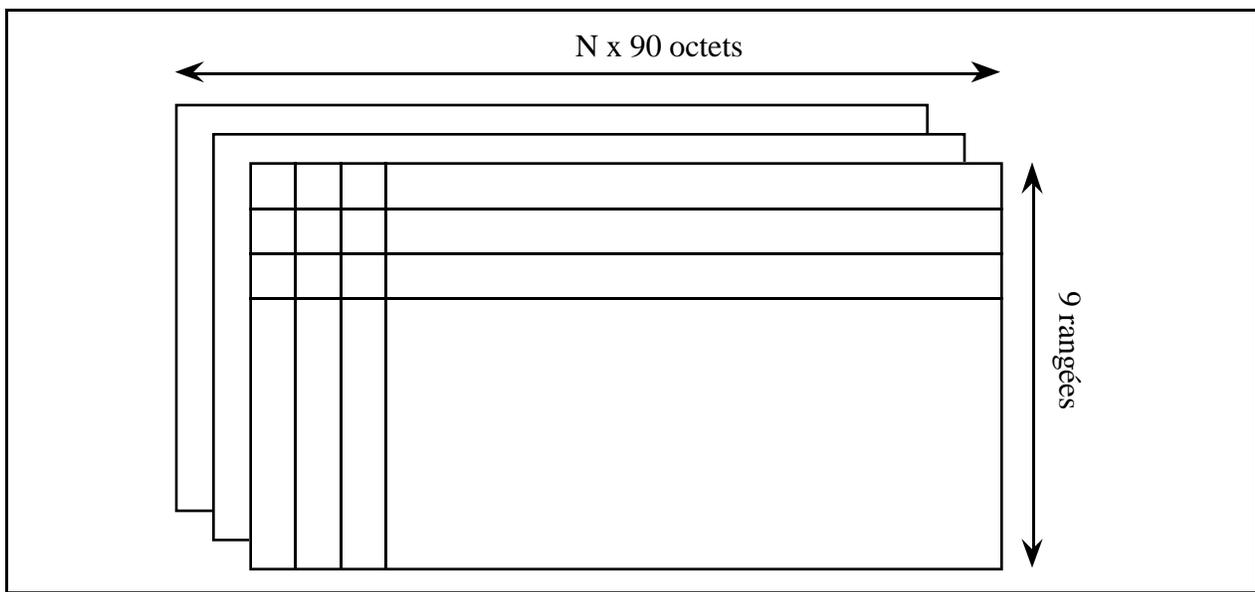


Figure B.6 : La trame SONET STS-N

D'un autre côté, l'UIT-T a adopté en 1988 une liste de recommandations concernant la hiérarchie digitale synchrone SDH [G.707, G.708, G.709]. SDH est un ensemble hiérarchisé de structures de transport numériques. Dans le cas d'ATM, le train de cellules est inséré dans un conteneur C-4 (cf. figure B.7). Ce conteneur est composé de 9 rangées de 260 octets. Cette valeur n'étant pas un multiple de la taille des cellules, la dernière cellule peut-être à cheval sur deux conteneurs. A ce conteneur, un en-tête POH (Path OverHead) est associé. Les octets ont la signification suivante :

- J1 : vérification de trajet.
- B3 : contrôle d'erreur de trajet.
- C2 : étiquette de signal de trajet.
- G1 : signalisation d'erreur de trajet.
- H4 : indicateur de décalage de cellule. Il indique le début de la première cellule contenue dans le conteneur.

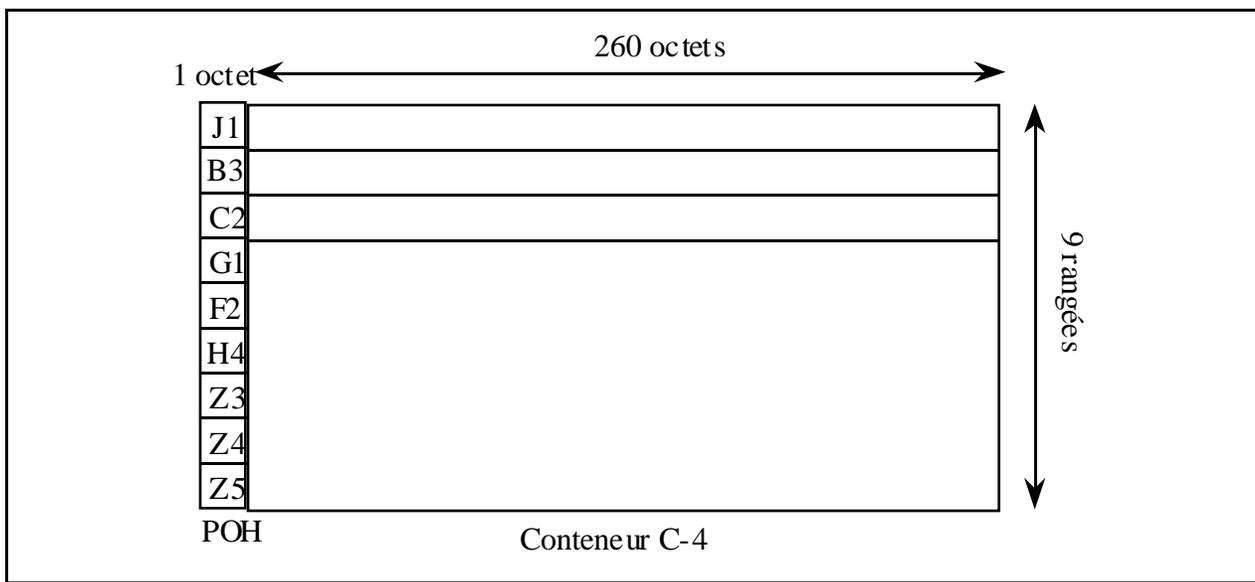


Figure B.7 : La structure du conteneur virtuel VC-4

La trame SDH de base appelée STM-1 (Synchronous Transport Module 1) est constituée du conteneur virtuel VC-4 auquel 9 octets ont été ajoutés sur chaque rangée (cf. figure B.8). Une trame STM-1 est émise toutes les 125 μ s, ce qui donne un débit de 155.52 Mbit/s. Le pointeur AU-4 de l'en-tête SOH indique la position du début d'un conteneur virtuel VC-4, puisque celui-ci peut être contenu dans deux trames.

Pour les débits de transmission plus élevés, N trames STM-1 sont multiplexées octet par octet pour former une trame STM-N. Les valeurs normalisées de N et les débits correspondants sont présentés dans la table B.1.

Pour obtenir des débits inférieurs, des conteneurs VC-1 (1,544 et 2,048 Mbit/s), VC-2 (6,784 Mbit/s) et VC-3 (34,368 et 44,736 Mbit/s) ont été normalisés. Ces trois conteneurs peuvent être multiplexés dans un conteneur VC-4.

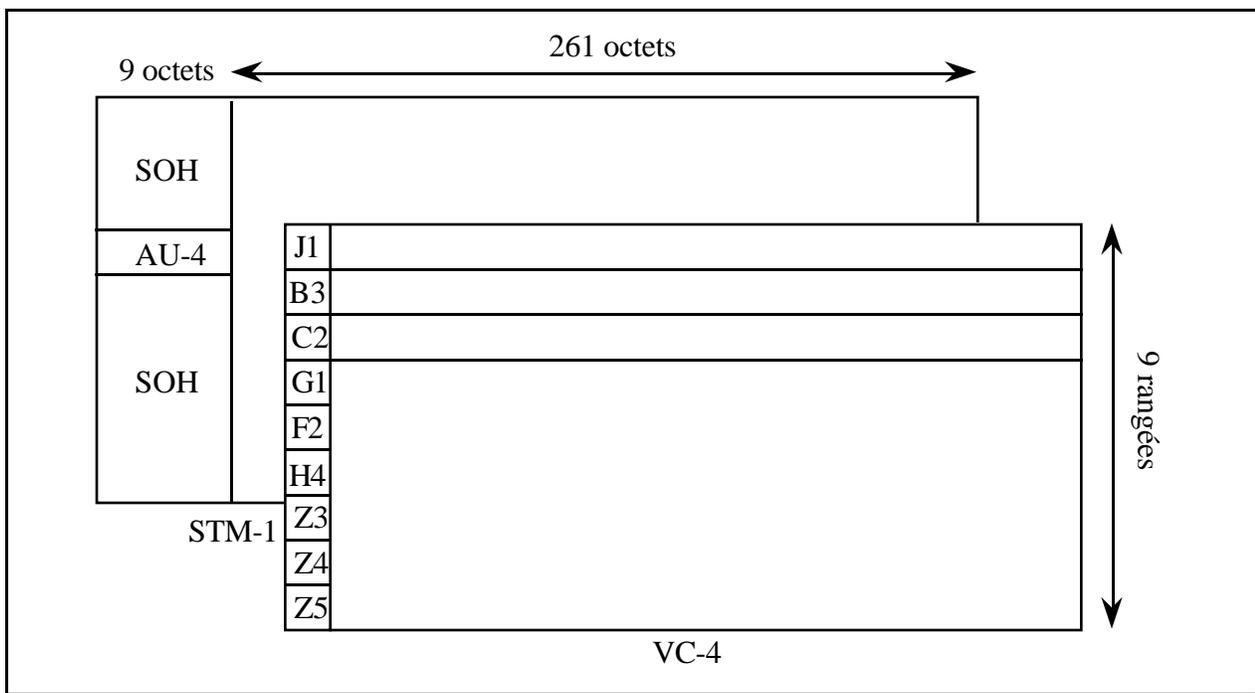


Figure B.8 : La structure de la trame STM-1

SDH	SONET	Débits (Mbit/s)
	STS-1	51,84
STM-1	STS-3	155,52
	STS-9	466,56
STM-4	STS-12	622,08
	STS-18	933,12
	STS-24	1 244,16
	STS-36	1 866,24
STM-16	STS-48	2 488,32

Table B.1 : Les trames SONET et SDH et les débits normalisés

Trois niveaux SDH offrent les mêmes débits que SONET (cf. table B.1). Les différences entre ces deux modes de transmission sont répertoriées dans un document de l'ANSI [Ans93].

B.1.6. La couche ATM

La couche ATM [I.361] est complètement indépendante du médium utilisé pour le transport des cellules ainsi que de la couche physique. Elle s'occupe principalement des

fonctions relatives aux traitements des champs de l'en-tête de la cellule ATM. On peut distinguer quatre fonctions principales :

- Le multiplexage et le démultiplexage des cellules des différentes connexions (identifiées par des valeurs différentes de VPI et VCI) au travers du flot de cellules.
- Le contrôle des paramètres de l'en-tête lors de la réception d'une cellule ou la génération de l'en-tête lors de l'émission d'une cellule.
- L'aiguillage des cellules dans les commutateurs en fonction des indicateurs de connexion VPI et VCI.
- Le mécanisme de contrôle de flux au niveau de l'interface usager-réseau grâce à l'utilisation du champ GFC.

La cellule ATM (cf. figure B.9) traitée dans cette couche se compose de 5 octets d'en-tête et de 48 octets de données utiles.

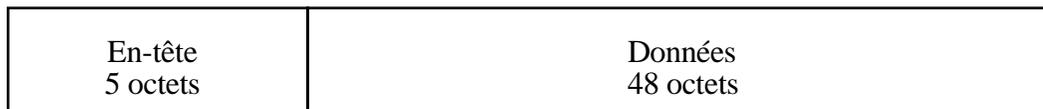


Figure B.9 : Le format de la cellule ATM

Au niveau de l'interface usager-réseau (UNI : User Network Interface), la structure de l'en-tête est représentée sur la figure B.10. Le premier champ contient 4 bits pour le GFC (Generic Flow Control).

Le second champ permet le routage de la cellule. Il comprend le champ VPI sur 8 bits et le champ VCI sur 16 bits.

Le champ PT (Payload type) est codé sur 2 bits et permet de spécifier le type des données transportées par la cellule. Certaines cellules servant à la gestion du réseau sont repérées par ce moyen. Ce champ vaut 0xx lorsqu'il s'agit de données utilisateurs, et 1xx s'il s'agit d'informations de gestion.

Le bit CLP (Cell Loss Priority) indique si la cellule a une priorité élevée (CLP = 0) ou si elle est sujette à élimination dans le réseau (CLP = 1) en cas de congestion.

Le bit RES (Reserved) pourra être défini par la suite suivant les besoins. Sa valeur par défaut est 0.

Le dernier champ de l'en-tête de la cellule est le HEC (Header Error Control) sur 8 bits. Il permet de vérifier l'absence d'erreur dans l'en-tête après la transmission à travers le réseau.

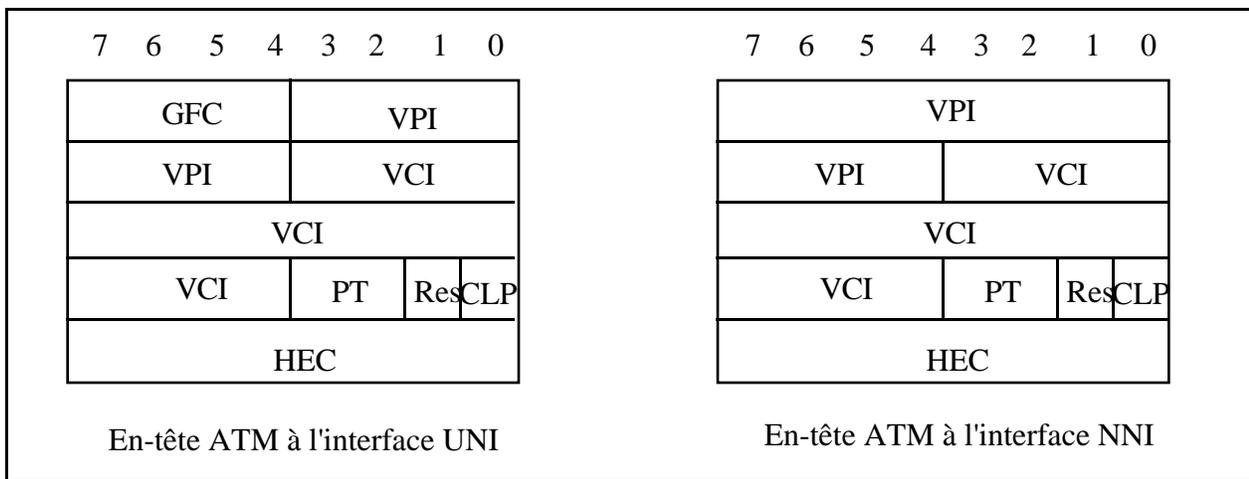


Figure B.10 : Les structures de l'en-tête de la cellule ATM

Au niveau de l'interface réseau-réseau (NNI : Network to Network Interface), l'en-tête de la cellule est légèrement modifié. Le champ GFC est supprimé et l'indicateur VPI a une taille de 12 bits.

B.1.7. La couche d'adaptation à ATM (AAL)

La couche d'adaptation à ATM (AAL : ATM Adaptation Layer) est nécessaire pour assurer la qualité de service requise pour les informations transportées. Ces services sont triés en 4 classes [I.362], chaque classe requérant un service particulier envers l'AAL [Cas92]. Pour identifier ces classes, trois paramètres ont été choisis :

- La synchronisation entre la source et la destination : certains services nécessitent une telle relation entre la source et la destination de l'information, d'autres non. Par exemple, pour le transfert de la voix à 64 kbit/s, il y a typiquement une synchronisation entre l'émetteur et le récepteur. Par contre, pour le transfert de données entre ordinateurs, il n'y a pas besoin d'une telle relation de temps.
- Le débit binaire : certains services ont un débit binaire constant (CBR : Constant Bit Rate), d'autres ont un débit binaire variable (VBR : Variable Bit Rate).
- Le mode de connexion : les services peuvent être soit en mode connecté, soit en mode non connecté [F.811, F.812].

Les quatre classes de service sont répertoriées dans la table B.2.

La couche AAL est divisée en deux sous-couche :

- La sous-couche de Segmentation et Réassemblage (SAR : Segmentation And Reassembly) s'occupe principalement de la segmentation des unités de données en cellules ATM lors de l'envoi, et du réassemblage des cellules multiplexées pour reconstituer les unités de données.
- La sous-couche de Convergence (CS : Convergence Sublayer) dépend du service requis.

	Classe A	Classe B	Classe C	Classe D
Synchronisation entre source et destination	Oui		Non	
Débit binaire	Constant	Variable		
Mode de connexion	Connecté			Non connecté
Applications types	Audio et vidéo à débit constant	Audio et vidéo à débit variable	Données en mode connecté	Données en mode non connecté

Table B.2 : Les classes de services de l'AAL

Pour les services supportés, il a été défini quatre types d'AAL [I.363]. L'AAL de type 1 est utilisée pour les services de classe A tels le transport du son de haute qualité, de l'image à débit constant et du téléphone.

L'AAL de type 2 correspond aux services de classe B.

L'AAL de type 3/4 concerne les services de classe C et D. Le nom de cette AAL provient du fait qu'à l'origine deux types distincts avaient été définis : l'AAL 3 pour les services de classe C et l'AAL 4 pour les services de classe D. Durant la normalisation de ces deux AAL, il est apparu qu'elles avaient des fonctions similaires. Le CCITT a alors décidé de regrouper ces deux AAL en une seule.

Le quatrième et dernier type d'AAL a été défini plus tardivement que les précédents. Il est nommé type 5 et concerne le transport des données. Il est fortement inspiré de l'AAL 3/4, mais en ne gardant que les parties indispensables. C'est une AAL simplifiée pour des raisons d'efficacité, en mode connecté.

B.1.7.1. AAL 1

Sous ce type sont transportées des informations à débit binaire constant ainsi que des informations assurant la relation temporelle entre l'émetteur et le récepteur.

La sous-couche SAR fournit les fonctions de segmentation et réassemblage au sens strict du terme, en utilisant un champ de 8 bits de l'unité de données qui comporte 48 octets à ce niveau (53 octets de la cellule auxquels on a enlevé les 5 octets de l'en-tête) (cf. figure B.11).

Les quatre premiers bits de ce champ contiennent un indicateur de séquençement (SN : Sequence Number) permettant de détecter la perte ou le déséquençement des cellules. Ce champ est scindé en deux parties, le bit CSI (Convergence Sublayer Indication) et les 3 bits du champ de décompte de séquence SNC (Sequence Number Counter).

Le bit CSI permet de transporter une marque de temps RTS (Residual Time Stamp) pour cadrer l'horloge du récepteur. Le calcul est effectué par évaluation de la différence de fréquence entre une horloge de référence commune extraite du réseau et une horloge de service. C'est la technique d'horodatage résiduel synchrone SRTS (Synchronous Residual Time Stamp). La marque de temps RTS comporte 4 bits, et est transportée en série dans le bit CSI d'une cellule sur deux.

Le champ SNC numérote les cellules sur 3 bits. Il est bien évident que cette numérotation n'est pas suffisante dans l'absolu, mais on suppose que la perte de deux cellules ou plus consécutives est extrêmement rare.

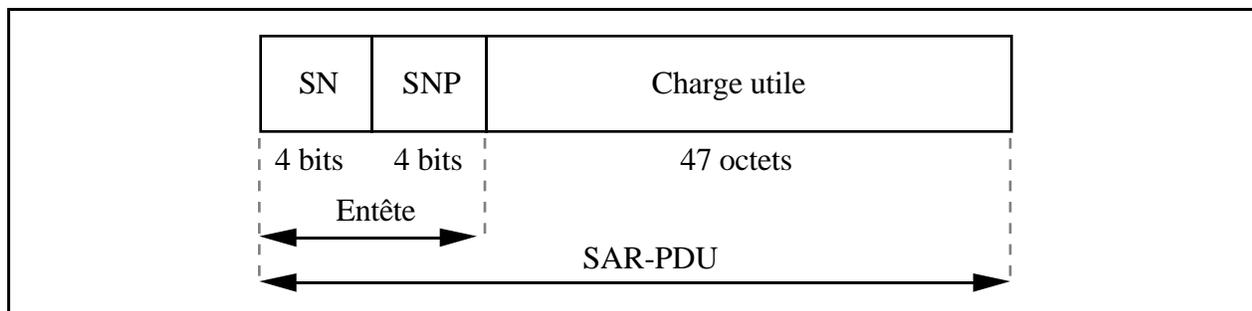


Figure B.11 : Le format du SAR-PDU pour l'AAL de type 1

Le champ SN est protégé par le champ SNP (Sequence Number Protection) de 4 bits. Le champ SNP est composé de la valeur du CRC sur 3 bits du champ SN, et d'un bit de parité sur les 7 bits des champs SN et CRC.

La sous-couche CS exécute les fonctions suivantes :

- La récupération de la variation de délai de l'arrivée des cellules pour fournir un débit binaire constant à l'utilisateur.
- Le transfert d'informations de temps.
- La détection d'erreur et la récupération de celles-ci.
- La génération de rapport sur l'état de la connexion entre les usagers.

B.1.7.2. AAL 2

L'AAL 2 traite les données à débit variable (VBR), en mode connecté, et nécessitant une synchronisation importante entre l'émetteur et le récepteur. Les fonctions sont assez semblables à celle de l'AAL 1. Cependant, la définition de cette classe de service est très peu avancée au sein de l'UIT-T.

B.1.7.3. AAL 3/4

L'AAL 3/4 traite les données en mode connecté et non connecté. Les données nécessitent un contrôle beaucoup plus strict du point de vue des erreurs. Les unités de données de ce type d'AAL ont donc un nombre plus important de champs de contrôle.

La couche AAL 3/4 est divisée en trois sous-parties : SAR (Segmentation and Reassembly), CPCS (Common Part Convergence Sublayer) et SSCS (Service Specific Convergence Sublayer). Les deux dernières couches correspondent à la sous-couche CS (cf. figure B.12). Pour le service en mode sans connexion du réseau public, la sous-couche SSCS est nulle.

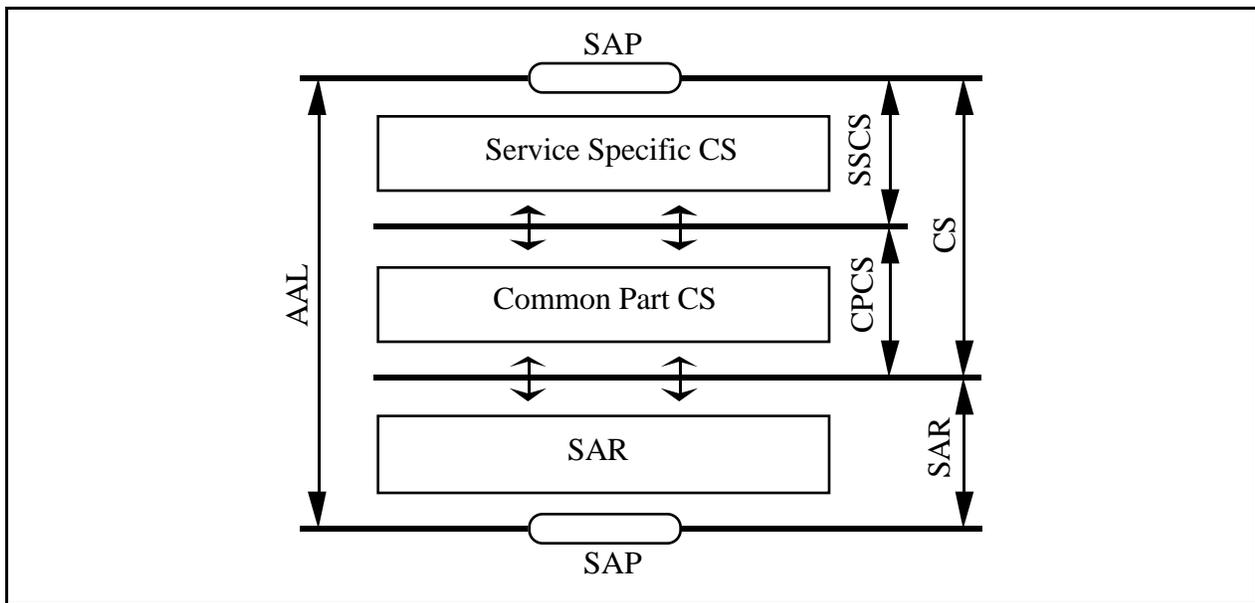


Figure B.12 : Les sous-couches de l'AAL 3/4

La sous-couche SAR exécute, outre les fonctions de segmentation et de réassemblage, de nombreux contrôles d'erreurs survenant sur la cellule. L'unité de données est représentée sur la figure B.13. Les champs ST (SAR Type) et SN (Sequence Number) permettent de vérifier le séquençement des cellules appartenant à une même connexion.

Le champ ST peut prendre les valeurs BOM (Beginning Of Message) pour la première cellule du message, EOM (End Of Message) pour la dernière cellule du message et COM (Continuation Of Message) pour les cellules intermédiaires. Une quatrième valeur nommée SSM (Single Segment Message) correspond à une cellule contenant la totalité d'un message.

Les cellules constituant un message ont des valeurs de SN consécutives. Ce champ permet de vérifier la perte ou l'insertion de cellules dans un message, puisque le protocole ATM conserve l'ordre des cellules sur les connexions.

Le champ LI (Length Indicator) indique la longueur des données dans la dernière cellule d'un message, c'est à dire une cellule ayant dans son champ ST les valeurs SSM ou EOM.

Le champ MID (Multiplexing Identifier) permet d'ajouter un niveau de multiplexage. Les cellules sont multiplexées au niveau ATM grâce au VP et VC, le champ MID fournissant une fonctionnalité supplémentaire au niveau de la couche AAL.

Enfin, le CRC (Cyclic Redundancy Check) permet de vérifier l'intégrité de la cellule. Il est calculé sur les 47 premiers octets du SAR-PDU.

La segmentation des messages en SAR-PDU de taille 48 octets s'opère comme décrit sur la figure B.14. Le message est découpé en paquets de 44 octets auxquels sont ajoutés les quatre octets de préfixe et suffixe de niveau SAR. Le dernier paquet n'étant pas forcément de taille 44 octets, le champ LI du SAR-PDU est rempli avec la longueur effective du paquet.

Lors de la réception des cellules, le réassemblage des messages est effectué. C'est le processus inverse de la segmentation.

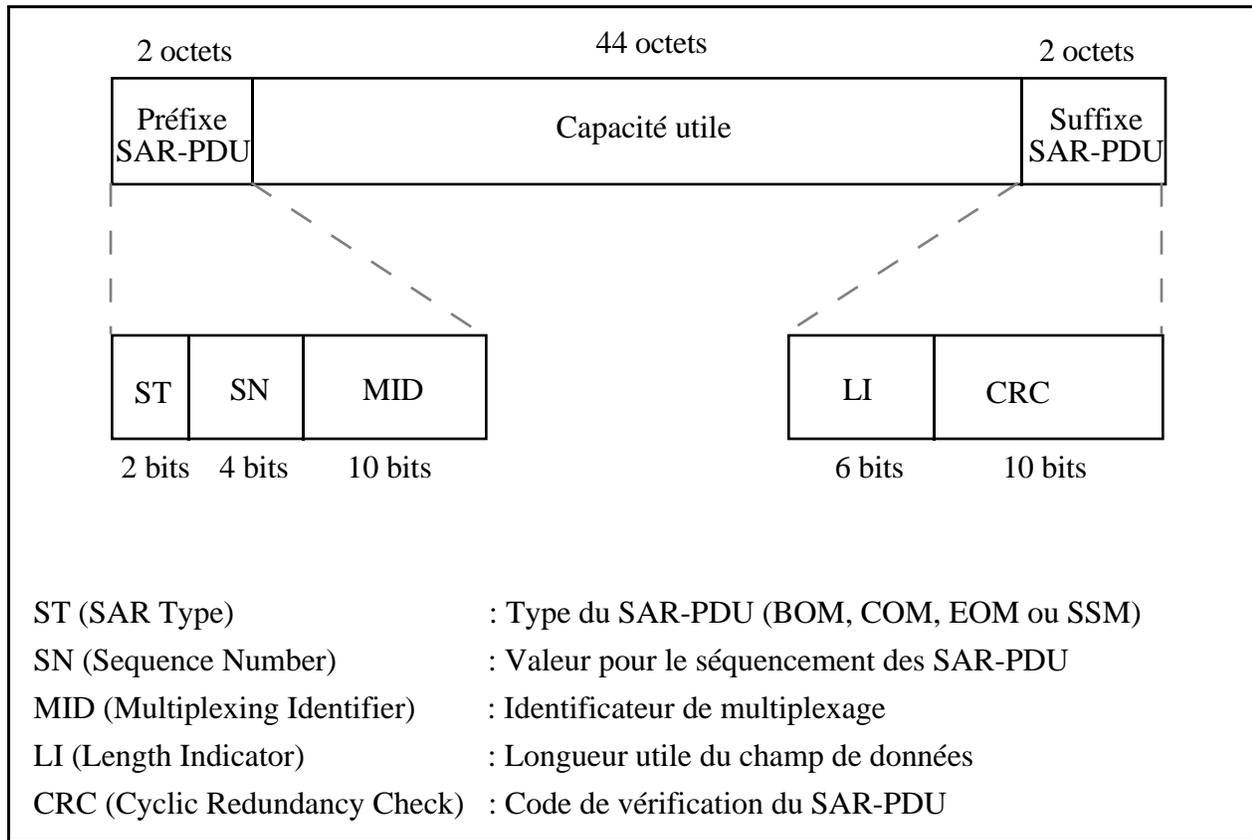


Figure B.13 : La structure du SAR-PDU pour l'AAL 3/4

La sous-couche CPCS travaille sur des messages de taille variable. Les informations de contrôle contenues dans le CPCS-PDU (cf. figure B.15) permettent un niveau de vérification supplémentaire.

Les champs Btag et Etag doivent être égaux pour un message donné. Ceci permet de vérifier l'absence de perte de cellules à une grande échelle. En effet, la détection de la perte de cellules effectuée dans la sous-couche SAR peut être insuffisante si un grand nombre de cellules est perdu. Par exemple, si toutes les cellules comprises entre une cellule ayant un champ SN de la sous-couche SAR égal à 5 et appartenant à un message dont le champ Btag est égal à 2 et une cellule ayant un champ SN égal à 6 appartenant à

un message différent du précédent dont le champ Etag est aussi égal à 3, alors le couple Btag-Etag détecte la perte de cellule. Cette perte d'un grand nombre de cellules est très peu probable mais elle est tout de même possible.

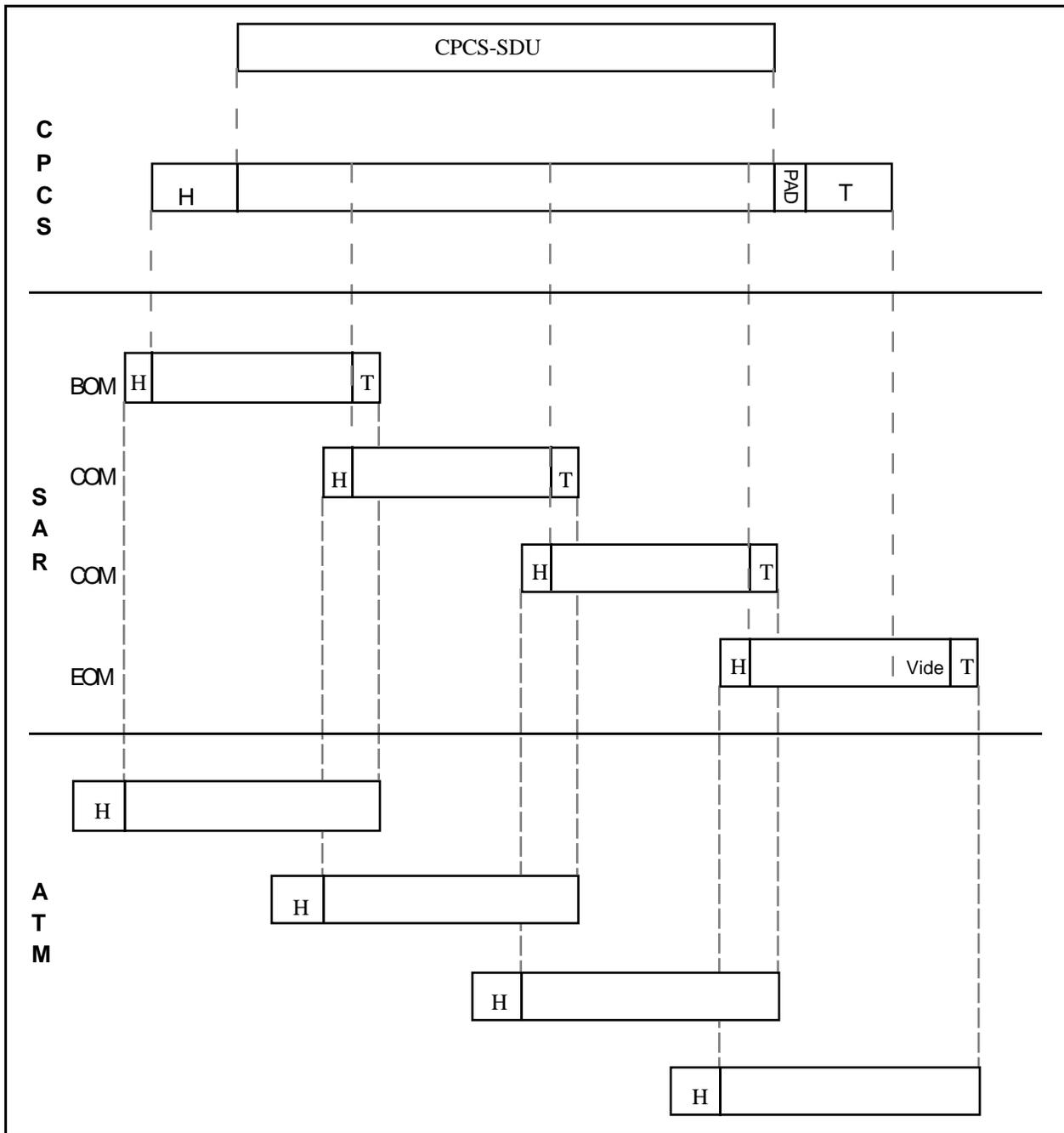


Figure B.14 : Du message à la cellule

Le champ BAsize contient une valeur supérieure ou égale à la capacité utile du message. Il permet de réserver la mémoire pour réassembler le message.

Le champ AL permet l'alignement du suffixe sur 32 bits, et ne transporte aucune information.

ATM-layer-user (AAU), est situé dans le champ PT de l'en-tête ATM. Une valeur égale à 1 indique la dernière cellule du message. Une valeur égale à 0 indique que la cellule contient le début ou la suite du message.

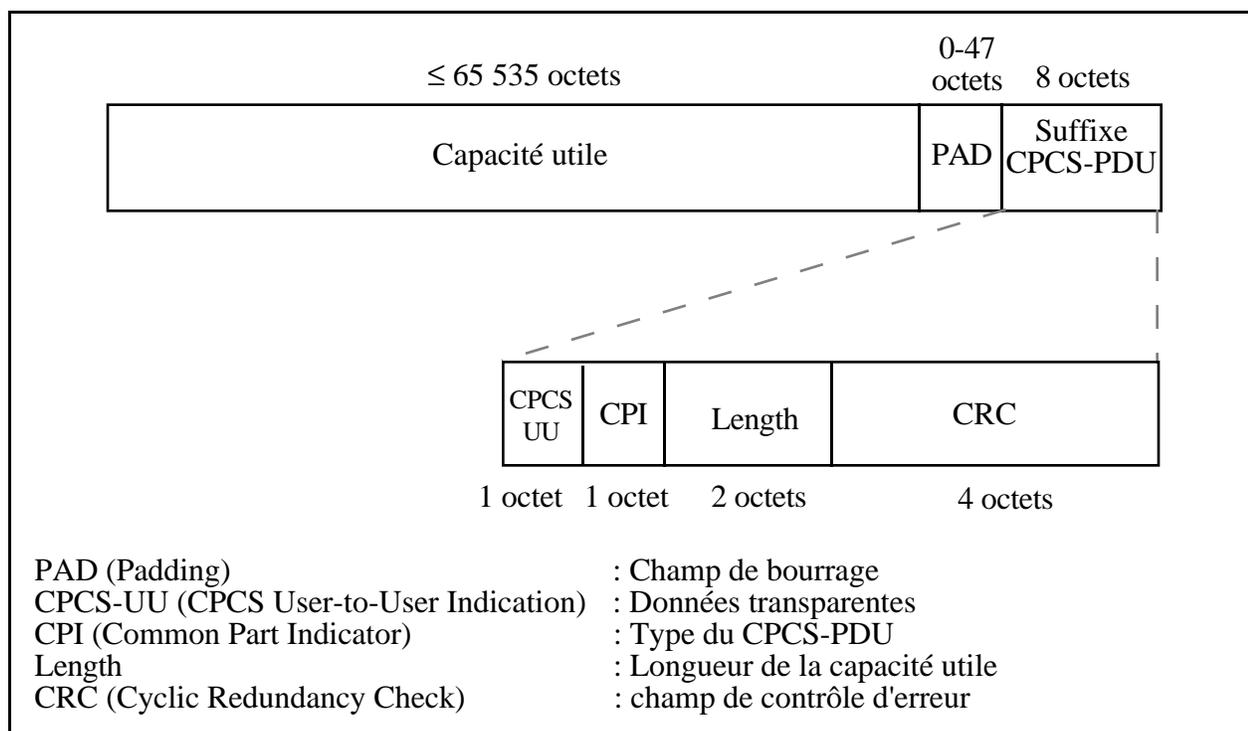


Figure B.16 : La structure du CPCS-PDU pour l'AAL 5

La sous-couche CPCS effectue un nombre plus important de fonctions que la précédente. Elle détecte les erreurs (erreur sur un bit, perte ou gain de cellule) et vérifie l'intégrité du message sur chaque connexion. Elle permet de transférer des données utilisateur dont la taille varie entre 1 et 65 535 octets. Elle doit cadrer cette taille sur un multiple de 48 octets. Le champ de bourrage a été placé à cet effet. Pour réaliser ces fonctions, un suffixe de 8 octets a été placé dans l'unité de données (cf. figure B.16).

Le champ CPCS-UU (CPCS User-to-User Indication) est utilisé pour transporter de manière transparente des données de niveau CPCS entre utilisateurs.

Une des fonctions retenue pour l'indicateur CPI (Common Part Indicator) est l'alignement sur 64 bits du suffixe du CPCS-PDU. Dans ce cas, le champ est rempli par des zéros. D'autres utilisations de ce champ sont à l'étude.

Le champ suivant donne la longueur totale du message en octets.

Le dernier champ (CRC : Cyclic Redundancy Check) est calculé sur l'ensemble du CPCS-PDU, c'est à dire sur la capacité utile, le champ de bourrage et les 4 premiers octets du suffixe. Il permet de détecter les erreurs de transmission.

B.1.8. Le service en mode sans connexion

Pour créer un mode sans connexion au-dessus de l'infrastructure connecté d'ATM, une couche CLNAP (ConnectionLess Network Access Protocol) [I.364] a été définie. Elle se place au-dessus de l'AAL 3/4.

La couche CLNAP s'occupe des problèmes d'adressage et de la sélection de la qualité de service. Pour se faire, une trame CLNAP comporte dans son en-tête les adresses source et destination au format E.164 (cf. figure B.17). Cette couche s'occupe aussi bien du routage que de la gestion des droits d'accès des utilisateurs en fonction des adresses présentes dans la trame.

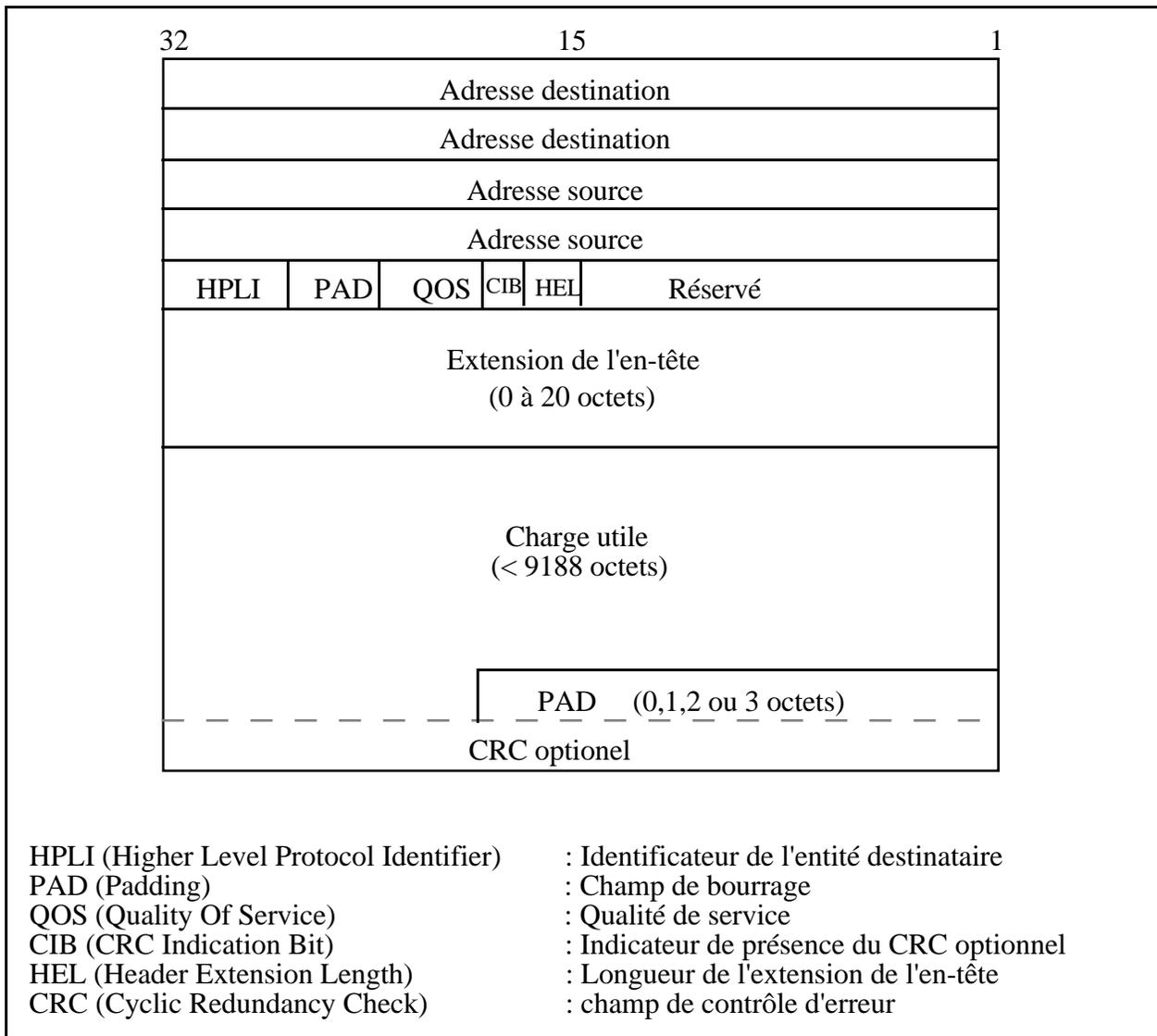


Figure B.17 : La structure de la trame CLNAP

Les adresses source et destination, au format E.164 (64 bits), permettent d'effectuer le routage de la trame sur le réseau ATM.

Le champ HLPI sur 6 bits sert à identifier l'entité de niveau utilisateur à laquelle la trame sera transmise au nœud de destination.

Le champ PAD est un champ de bourrage : il est rempli par des 0 pour que la longueur de la trame soit multiple de 4 octets et puisse ainsi être traitée par un processeur 32 bits.

Le champ QOS permet d'indiquer la qualité de service requise pour le CLNAP-PDU. Les valeurs pour ce champ ne sont pas encore définies.

Le champ CIB indique la présence (CIB = 1) ou l'absence (CIB = 0) du CRC.

Le champ HEL indique la longueur du champ d'extension de l'en-tête.

Le champ Extension de l'en-tête n'est pas utilisé. Il pourra servir par la suite à faire circuler des informations de service en même temps que des trames.

Le CRC est optionnel car dans la plupart des cas, il n'est pas nécessaire d'opérer une telle vérification puisque des vérifications de ce type sont effectués dans les couches inférieures.

B.1.9. Les principes d'exploitation et de maintenance

Les principes d'exploitation et de maintenance du RNIS à Large Bande sont définis dans la recommandation I.610. Ils s'intéressent à la gestion de l'accès au réseau, pour les couches Physique et ATM. Les opérations de contrôle et de maintenance (OAM : Operation And Maintenance) sont les suivantes :

- Le contrôle de la qualité de fonctionnement.
- La détection des fautes et des dérangements.
- La protection des systèmes, par retrait d'une entité en dérangement.
- La fourniture des informations de dérangement et de performance.
- La localisation des dérangements.

Les fonctions OAM sont classées en cinq niveaux hiérarchiques associés au deux niveaux Physique et ATM du modèle de référence (cf. figure B.18). Elles sont réalisées par des flux d'informations bidirectionnels. Les niveaux sont les suivants :

- Le niveau des voies virtuelles (flux F5) surveille l'état des connexions (VC).
- Le niveau des conduits virtuels (flux F4) surveille l'états des chemins (VP).
- Le niveau des conduits de transmission (flux F3) surveille les liens physiques.

- Le niveau des sections numériques (flux F2) surveille l'état des sections physiques (par exemple, trois sections STS-3 à 51,84 Mbit/s dans un lien SONET à 155,52 Mbit/s).
- Le niveau des sections élémentaires régénérées, parties d'une section numérique (flux F1).

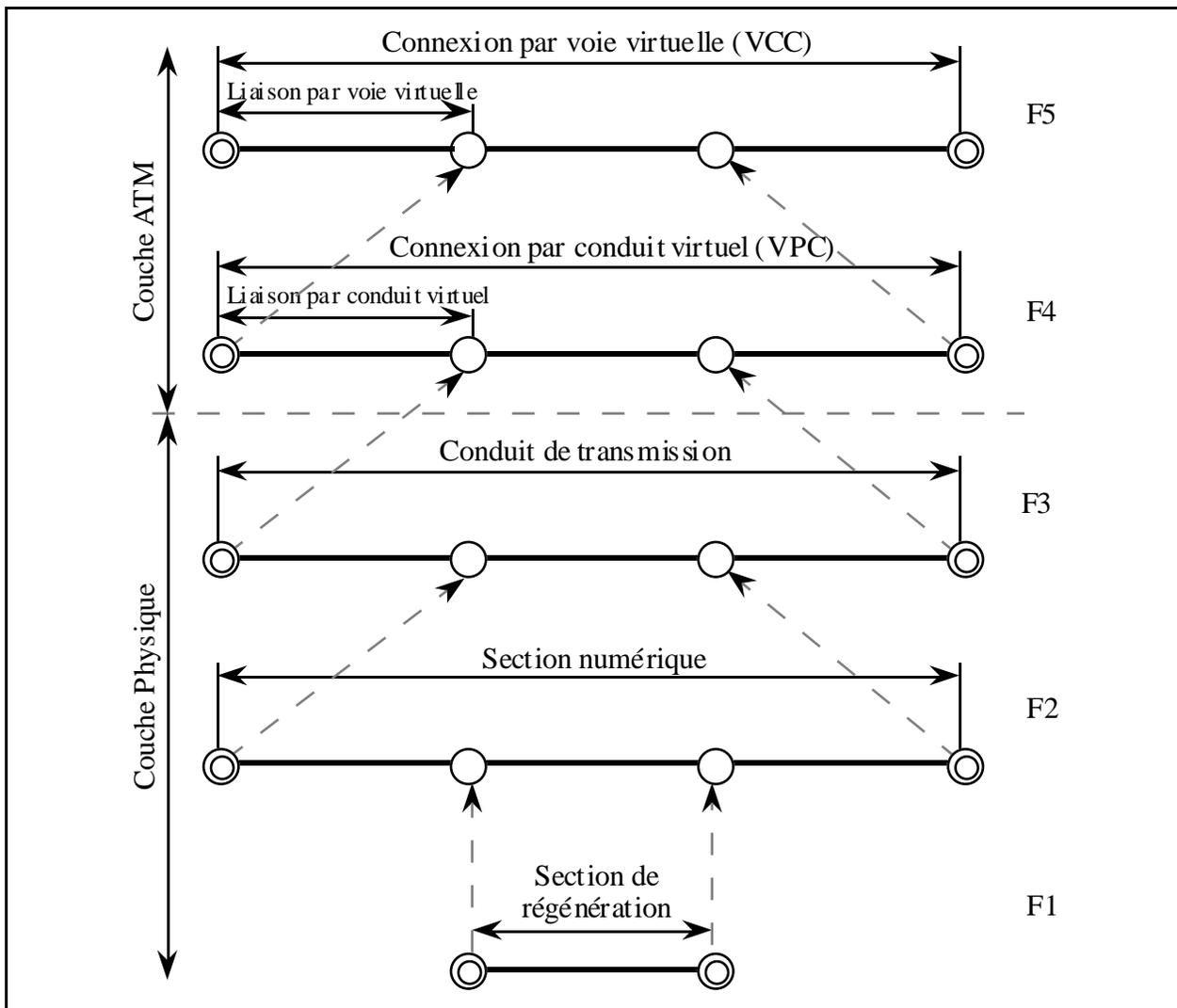


Figure B.18 : Les niveaux hiérarchiques OAM

Les flux F1, F2 et F3 correspondent à la couche Physique et permettent de gérer les liens reliant les nœuds du système. Les deux premiers sont insérés dans le champ SOH dans le cas d'une transmission SDH. Le flux F3 est inséré dans le champ POH.

Les flux F4 et F5 correspondent à la couche ATM et permettent de contrôler l'état des chemins virtuels et des voies virtuelles. Ils sont insérés dans le trafic ATM. Les flux F4 ont une valeur du VPI pré-assignée. Les flux F5 sont identifiés par une valeur particulière du champ PT de l'en-tête de la cellule.

Chaque niveau est indépendant du niveau précédent.

B.1.10. La signalisation ATM

La signalisation gère l'établissement, le maintien et la libération des connexions ATM. La mise en place d'un chemin virtuel s'effectue par l'intermédiaire d'un plan spécifique, le plan de contrôle du modèle de référence du protocole ATM (cf. figure B.4), contrairement à des protocoles comme X.25 où la mise en place d'un circuit virtuel se fait par l'intermédiaire d'un paquet d'appel passant par le plan utilisateur.

Deux solutions ont été retenues pour transmettre les éléments de signalisation sur le réseau, ces deux solutions pouvant cohabiter :

- Une signalisation par canal sémaphore.
- Une signalisation dans la bande.

Dans le premier cas, des connexions virtuelles spécifiques sont réservées. Elles sont créées par une procédure appelée métasignalisation. Cette dernière a pour but d'établir, de libérer et de maintenir les connexions de signalisation (SVC : Signalling Virtual Channel), de résoudre les problèmes d'attribution des VPI/VCI et de prendre en charge la gestion de la bande allouée par les SVC.

Pour les connexions de signalisation, une couche AAL particulière a été définie, appelée SAAL (Signalling ATM Adaptation Layer) [Q.2100]. Elle s'appuie sur l'AAL 5 jusqu'au niveau CPCS. Par contre, la sous-couche SSCS est particulière. Elle est décomposée en deux parties :

- SSCOP (Service Specific Connection Oriented Protocol) qui offre un service de transfert assuré [Q.2110].
- SSCF (Service Specific Coordination Function) qui offre le service nécessaire au protocole de signalisation. Plusieurs sous-couches SSCF sont définies : la recommandation Q.2130 pour la signalisation entre l'utilisateur et le réseau (UNI), la recommandation Q.2140 pour la signalisation entre les nœuds du réseau (NNI).

Les protocoles de signalisation sont nombreux. La signalisation entre l'utilisateur et le réseau est définie dans [Q.2931] par l'UIT-T. Une spécification en a été dérivée par l'ATM Forum. La signalisation entre nœuds du réseau NNI est en cours de spécification par l'ATM Forum. L'organisation de ces protocoles est représentée sur la figure B.19.

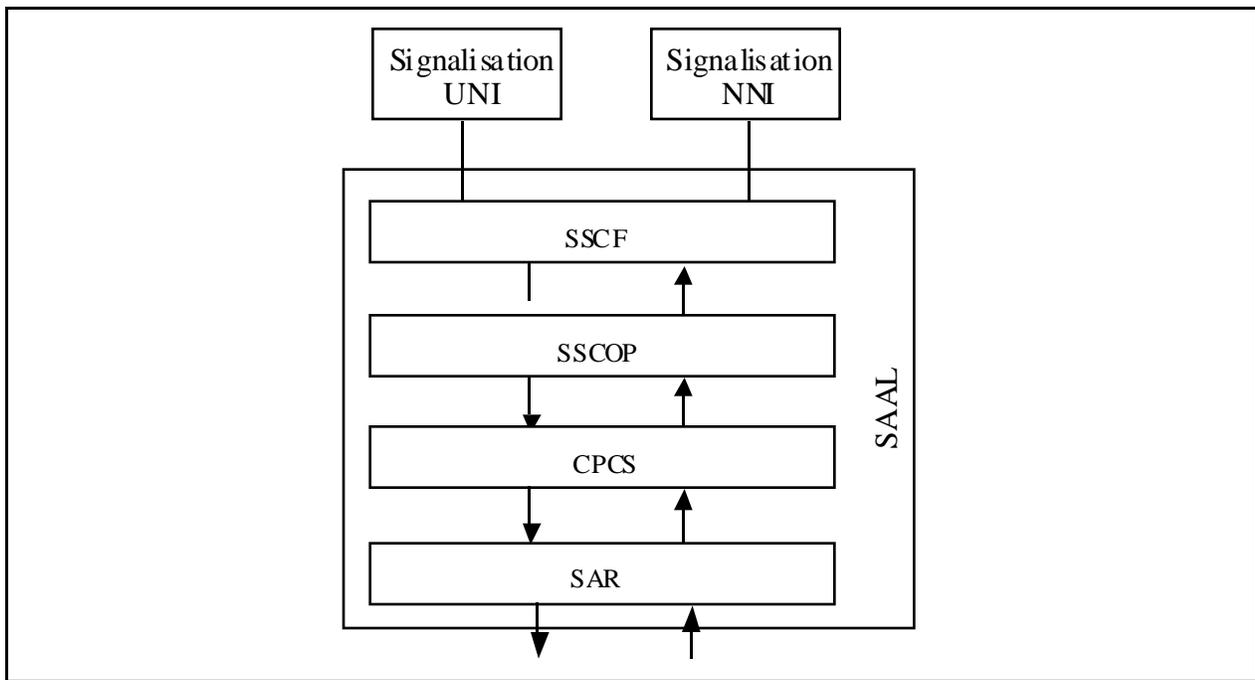


Figure B.19 : L'organisation des protocoles pour la signalisation par canal sémaphore

La seconde solution, la signalisation dans la bande, est mieux adaptée à la gestion dynamique des ressources du réseau. Les cellules sont identifiées par une valeur réservée du champ PT de l'en-tête. Le contrôle de flux pourra être assuré par ce type de signalisation.

B.1.11. La gestion du trafic et des encombrements dans les réseaux ATM

La gestion du trafic et des encombrements dans les réseaux ATM fait l'objet de la recommandation I.371. Les fonctions nécessaires peuvent être divisées en cinq grandes branches :

- La gestion des ressources (NRM : Network Resource Management) permet de prévoir l'attribution des ressources de réseau de manière à séparer les flux des différents services.
- La fonction d'admission et de contrôle (CAC : Connection Admission Control) permet d'accepter ou de refuser l'ouverture d'une connexion, ou la renégociation de sa qualité de service. Le routage est partie intégrante de cette fonction.
- Le contrôle des paramètres d'utilisation (UPC : Usage Parameter Control) ou de réseau (NPC : Network Parameter Control) qui permet de surveiller et gérer le trafic en termes de trafic offert et de validité de connexion ATM, respectivement au niveau de l'accès usager et de l'accès réseau. Le but de cette fonction est de

protéger les ressources du réseau contre les actes malveillants ou les erreurs involontaires pouvant dégrader la qualité de service des autres connexions. Par exemple, un usager ayant négocié un débit crête de 10 Mbit/s et envoyant 100 Mbit/s sur le réseau devra être sanctionné. Une méthode est de supprimer les cellules en surplus.

- Les commandes de rétroaction regroupent l'ensemble des actions exécutées par le réseau et par les usagers pour réguler le trafic en fonction de l'état du réseau.
- La gestion des priorités (PC : Priority Control) qui permet à un usager de générer des flux avec différentes priorités (bit de priorité à la perte contenu dans l'en-tête de la cellule ATM). Ainsi, un élément de réseau encombré peut rejeter sélectivement les cellules à faible priorité pour préserver la qualité de service des autres trafics.

Ces fonctions de gestion prennent place à différents endroits du réseau. La figure B.20 montre leur répartition dans un réseau ATM. D'autres fonctions pourront être ajoutées par la suite.

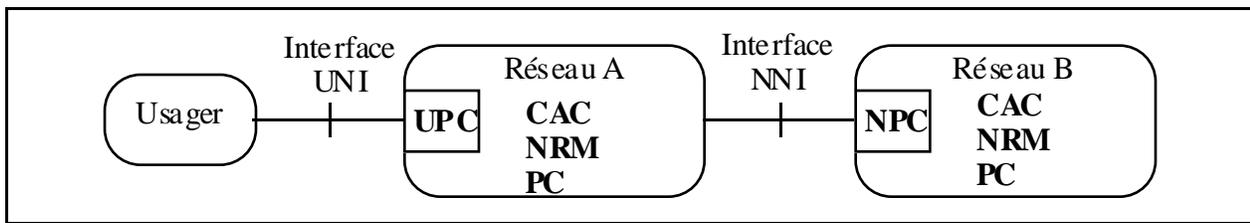


Figure B.20 : Configuration de référence pour la gestion du trafic et des encombrements

B.1.11.1. La gestion des ressources (NRM)

L'allocation des ressources pour un réseau ATM repose essentiellement sur la notion de chemin virtuel, et plus particulièrement les connexions de conduits virtuels (VPC) car elles permettent :

- De simplifier la commande d'admission et de contrôle (CAC).
- De gérer des priorités par séparation des types de trafics en fonction de leur qualité de service.
- De distribuer efficacement les messages relatifs à la mise en œuvre des schémas de gestion du trafic (par exemple pour indiquer l'encombrement dans le réseau en distribuant un message unique à toutes les connexions de voies virtuelles (VCC) composant une connexion VPC).

- De regrouper les services usager-usager ou réseau de manière à simplifier les commandes des fonctions UPC/NPC pour qu'elles puissent être appliquées globalement au trafic composite.

Les méthodes d'allocation de ressources définies dans la recommandation I.371 s'appuie uniquement sur le débit crête des connexions. Elles permettent effectivement de garantir les qualités de service requises, mais conduisent à une sous-utilisation du réseau.

Mais cette fonction restera sans doute spécifique à chaque opérateur. Des mécanismes d'allocation rapide des ressources (FRM : Fast Resource Management) sont aussi apparus. Deux versions du protocole ont été proposées par le CNET [Boy92, Gui92] : une version négociée avec transmission retardée mais garantie (FRP/DT : Fast Reservation Protocol Delayed Transmission) et une version avec transmission immédiate (FRP/IT : Fast Reservation Protocol Immediate Transmission).

Avec le protocole FRP/DT, une source demande une augmentation ou une diminution de son débit crête et attend que la réservation soit effectuée pour changer son activité. Si les ressources sont insuffisantes, la source est bloquée.

Dans le protocole FRP/IT, l'allocation d'un nouveau débit crête est effectuée de proche en proche dans chacun des éléments du réseau le long de la route de la connexion. Le nouveau débit est accepté à l'accès dès que la requête est conforme au contrat de trafic, sans attendre l'allocation dans les éléments du réseau. Ce type de réservation de ressources est utile pour les sources à débit variable émettant des rafales, par exemple les services sans connexion ou certains flux vidéo compressés.

B.1.11.2. La fonction d'admission et de contrôle (CAC)

La fonction d'admission des connexions est définie comme étant l'ensemble des actions exécutées par le réseau au cours de la phase d'établissement de l'appel, ou au cours de la renégociation afin d'établir si une nouvelle connexion sera acceptée ou rejetée. Elle contrôle l'accès des services au réseau et s'assure que des ressources suffisantes sont disponibles. Elle nécessite au minimum deux informations :

- Les descripteurs de trafic source.
- La classe de qualité de service requise.

La commande d'admission et de contrôle utilise ces informations pour accepter ou non la connexion et pour déterminer l'acheminement et l'attribution des ressources du réseau.

Le réseau ATM devant gérer des trafics avec des exigences de qualité de service diverses, il est difficile de concevoir des méthodes CAC efficaces. La méthode la plus simple pour résoudre ce problème est de classer les sources de trafic en fonction des exigences de qualité de service. Les ressources sont allouées à chaque classe et sont contrôlées de manière indépendante.

B.1.11.3. Le contrôle des paramètres d'utilisation ou de réseau (UPC/NPC)

L'objectif principal de ce contrôle est de protéger les ressources du réseau contre les intentions malveillantes des utilisateurs, qui pourraient affecter la qualité de service des autres connexions. Ce contrôle est assuré par des mécanismes de surveillance chargés de détecter les violations de contrat et de prendre les mesures appropriées. Il vérifie aussi la validité des VPI et VCI, en particulier s'ils sont ou non attribués.

La recommandation I.371 définit une commande UPC/NPC au niveau de la cellule, et basée uniquement sur le débit crête. Les actions possibles sont les suivantes :

- Laisser passer la cellule lorsque celle-ci est conforme.
- Reprogrammer la cellule, c'est à dire reformer le flux. Cette action est possible lorsqu'une méthode de conformation du trafic est combinée avec la commande UPC/NPC). Des exemples de ces méthodes seront présentées dans la suite de ce paragraphe.
- Marquer la cellule, c'est à dire mettre le bit CLP de la cellule à 1 lorsque celui-ci était à 0.
- Rejeter la cellule lorsque celle-ci n'est pas conforme.

En plus de ces quatre actions, la commande UPC/NPC peut aussi déclencher la libération de la connexion.

De nombreuses autres méthodes sont apparues pour la commande NPC/UPC ces dernières années. La suite de ce paragraphe résume les principales. Aucune n'est normalisée actuellement.

B.1.11.3.1. Les mécanismes de surveillance

Deux grands types de mécanismes de surveillance ont été proposés : les mécanismes à fenêtres et les Leaky Buckets (seau percé). De nombreuses variantes ont été proposées pour chacun des deux mécanismes. La suite de paragraphe en examine quelques-unes.

Les mécanismes à fenêtres sont scindés en deux : la fenêtre sautante et la fenêtre glissante. Un algorithme de fenêtre sautante observe un flux de cellules pendant une durée T. Lorsque cette période est écoulée, une nouvelle période d'observation (fenêtre) commence. Le flux est donc observé par tranche de période T. Dans chaque fenêtre, chaque connexion peut faire passer un nombre de cellules N, relatif au débit.

L'algorithme de fenêtre glissante consiste à observer le flux de cellules à travers une fenêtre qui se déplace de slot en slot. Le principe de contrôle est le même que pour la fenêtre sautante.

Une variation de la fenêtre sautante est l'algorithme EWMA (Exponentially Weighted Moving Average) qui observe un flux de cellule à travers une fenêtre sautante à l'intérieur de laquelle le nombre de cellules admissibles pour une connexion varie au cours du temps [Rat91].

Ces mécanismes à fenêtre se sont avérés peu efficaces, bien que la fenêtre glissante soit meilleure que la fenêtre sautante. Par conséquent, des mécanismes plus performants ont été proposés : les Leaky Buckets. La première version a été introduite par J. Turner [Tur86]. Dans ce mécanisme, les crédits sont engendrés de manière périodique à une vitesse dépendant du contrat de trafic. Les cellules qui arrivent et qui trouvent un crédit disponible accèdent au réseau. Chaque émission consomme un crédit. Par contre, une cellule qui arrive et qui ne trouve pas de crédit est rejeté. Ceci conduit à un nombre de perte important.

La première variante consiste à ajouter un tampon dans lequel les cellules seront stockées en attendant un crédit. De nombreuses autres variantes ont été proposées. Elles peuvent être trouvées dans [Lem93, Par94]. Les Leaky Buckets permettent de garantir que le débit moyen d'une connexion est respecté.

B.1.11.3.2. La conformation du trafic

La conformation du trafic agit sur les caractéristiques d'un flux de cellules sur une connexion VCC ou VPC pour les modifier dans le sens souhaité. Elle doit cependant veiller à préserver l'intégrité de séquençement des cellules sur la connexion ATM. De plus, le retard engendré par cette opération doit rester dans les limites négociée lors de l'ouverture de la connexion. Ce mécanisme s'ajoute à la commande UPC/NPC. C'est une option d'après la recommandation I.371.

Différentes actions sont possibles comme par exemple la réduction du débit crête, la limitation de la longueur des rafales, ou la réduction de la variation du temps de propagation des cellules par leur espacement dans le temps.

Le contrôleur-espaceur proposé par le CNET [Gul92] est une implantation de ces méthodes. Il agit sur le débit crête, et reforme le trafic en espaçant les cellules. La fonction d'espacement est réalisée en associant à chaque connexion un algorithme qui gère un échancier de réémission pour la connexion. Les cellules en excès dans le flux sont stockées jusqu'à ce qu'elles puissent être émises.

Des extensions de Leaky Buckets permettent d'assurer la conformation du trafic. Ainsi, le mécanisme Buffered Leaky Bucket [Sid89] fonctionne avec des crédits comme les Leaky Buckets classiques, mais dispose en plus d'un tampon. Ainsi, lorsqu'il n'y a pas de crédit disponible, une cellule qui arrive est stockée, et attend la disponibilité d'un crédit pour être émise.

B.1.11.4. Les commandes de rétroaction

Le contrôle de congestion réactif permet de répondre aux surcharges du réseau, causées en particulier par la superposition des trafics et les rafales. La recommandation I.371 définit le mécanisme de notification d'encombrement EFCI (Explicit Forward Congestion Indication). La fonction de celui-ci est de transporter les informations de congestion le long du conduit virtuel entre l'émetteur et le récepteur. Ceci se fait à l'aide du champ PT contenu dans l'en-tête de la cellule. Ainsi, une cellule passant par un nœud du réseau surchargé à son champ PT mis à 010 ou 011. La destination qui détecte des cellules avec ces valeurs dans le champ PT peut retourner l'information de congestion vers la source avec le mécanisme BCN (Backward Congestion Notification).

En raison des temps de propagation importants dans un réseau ATM, un mécanisme de notification d'encombrement directement vers l'émetteur a été proposé. Le mécanisme EBCN (Explicit Backward Congestion Notification) permet à un commutateur d'avertir tous les émetteurs qu'une congestion a lieu à son niveau. La notification passe par les flux OAM F5.

B.1.11.5. La gestion des priorités (PC)

Une source peut générer plusieurs flux de cellules en utilisant le bit CLP. Ainsi, deux qualités de services différentes peuvent être associées à ces flux. La recommandation I.371 définit deux types de contrats avec ce bit CLP.

Lorsque l'utilisateur génère uniquement un flux CLP=0, tout le trafic est accepté dans le réseau, mais une violation du contrat entraîne le rejet des cellules en excès.

L'utilisateur peut aussi définir deux qualités de service, l'une pour le flux CLP=0 et l'une pour le flux CLP=0+1. Les cellules non conformes du flux CLP=0 sont marquées. Par contre, les cellules non conformes du second flux sont immédiatement rejetées.

D'autres méthodes de priorité et de rejet sélectif ont été proposées. Dans ce cas, la priorité n'est plus transportée dans la cellule, mais est fixée pour la connexion lors de la configuration, et est stockée au niveau des nœuds du réseau.

Ainsi, la méthode HOL (Head Of Line) [Sai90] permet à chaque classe de cellules de disposer d'un délai maximal pour quitter le commutateur.

Une amélioration possible consiste à diviser les tampons en plusieurs files de priorités décroissantes. Le mécanisme HOL-PJ (Head Of Line with Priority Jump) [Lim88] permet à une cellule ayant séjourné pendant un certain temps dans une file de passer dans une file de priorité plus élevée.

D'autre part, la méthode QLT (Queue Length Threshold) donne la priorité aux cellules sensibles à la perte de données si le nombre de cellules dans le tampon d'entrée dépasse un certain seuil. Sinon, la priorité est donnée aux cellules sensibles au délai.

B.2. L'état de la normalisation ATM

La normalisation au niveau international du Réseau Numérique à Intégration de Services Large Bande a débuté en Janvier 1985 et est aujourd'hui étudiée par plusieurs organismes de normalisation. Elle fait suite à la normalisation du RNIS bande étroite (64 Kbit/s), plus connu sous le nom commercial de Numéris en France.

Le CCITT (Comité Consultatif International Télégraphique et Téléphonique), aujourd'hui renommé UIT-T, est à l'origine de la normalisation du RNIS Large Bande. De nombreux groupes de travail à l'intérieur du CCITT sont concernés par ces études. D'autres organismes tels que l'ISO (International Standard Organisation), l'ANSI (American National Standards Institute) aux Etats-Unis, l'ETSI (European Telecommunications Standards Institute) en Europe s'intéressent à la normalisation de la technique ATM. En 1991, l'ATM Forum a été créé. Cette organisation regroupe aujourd'hui plus de 500 industriels désirant accélérer le déploiement des produits et des services ATM en fournissant des spécifications sur lesquelles ils ont trouvé un consensus. L'IETF (Internet Engineering Task Force) s'intéresse aussi à ATM, en particulier pour le couplage d'IP sur ce mode de transfert.

La première recommandation du CCITT relative au RNIS-LB [I.121] est apparue en 1988. Elle définissait une base de travail pour le développement et la normalisation du

RNIS-LB. En Novembre 1990, le CCITT approuvait treize nouvelles recommandations établissant les principes et les spécifications initiales du RNIS-LB. Depuis, des travaux importants ont été effectués et de nombreuses améliorations ont été apportées.

Cependant, un grand nombre de points restent ouverts. Du point de vue du protocole, la couche physique est bien définie. La couche ATM a été spécifiée presque entièrement en 1992. Seule l'utilisation du champ GFC de la cellule ATM n'est pas clairement définie puisque plusieurs solutions sont en cours d'études. Les couches d'adaptation à ATM pour les types 1, 3/4 et 5 sont en grande partie définies. En revanche, un travail très faible a été fait sur l'AAL de type 2. Les premiers travaux ont d'ailleurs été remis en cause.

D'autre part, la signalisation sur le réseau ATM n'est pas complètement définie. Pour l'interface usager-réseau, la spécification UNI3.1 a été définie par l'ATM Forum en Janvier 1995. C'est une version simplifiée de la recommandation Q.2931, à laquelle a été ajouté l'établissement de connexions point à multipoints. Elle permet en particulier à un nœud principal d'ajouter un participant à une connexion point à multipoints, mais pas à un participant de s'ajouter à son initiative. Bien évidemment, il est possible d'établir des connexions permanentes ou semi-permanentes. Mais les paramètres de qualité de service, uniquement le débit actuellement, ne sont pas renégociables après la création d'une connexion. L'ATM Forum travaille aujourd'hui sur la version 4.0 de cette spécification, qui devrait être disponible dans la deuxième moitié de l'année 1995. Elle prendra en compte l'ajout d'un participant à une connexion point à multipoints à l'initiative de celui-ci. Par contre, il n'est pas certain qu'elle prenne en compte les connexions multipoints à multipoints. Elle permettra cependant la renégociation des qualités de service des connexions.

Pour la signalisation entre nœuds du réseau, il existe une ébauche de spécification établie par l'ATM Forum et appelée P-NNI (Private NNI) phase 0 ou IISP (Interim Inter-Switch Signalling Protocol). Cette dernière a pour but de définir une signalisation NNI pour les réseaux privés. La version 0 est une version intermédiaire en attendant la version 1, annoncée pour la deuxième moitié de l'année 1995. Elle utilise la signalisation UNI 3.1. Elle ne sera pas compatible avec la version 1, qui est très ambitieuse. Outre le support de la qualité de service et le contrôle d'admission (CAC), cette dernière est prévue pour s'adapter à toute taille de réseau (de quelques commutateurs à plusieurs millions de commutateurs) grâce à un protocole en arbre.

Les réseaux publics font l'objet d'études séparés. La signalisation NNI est étudiée dans un groupe de l'ATM Forum appelé B-ICI (Broadband Inter-Carrier Interface) et par l'UIT-T.

Du point de vue de la gestion du trafic, il existe aujourd'hui des mécanismes simples permettant de contrôler l'accès au réseau. Ils sont basés principalement sur le débit crête. Mais, les méthodes de contrôle font à l'heure actuelle l'objet de nombreuses recherches. La recommandation I.371 nécessite de nombreux compléments d'étude. Les paramètres de qualité de service restent à définir.

B.3. Les travaux de recherche sur ATM

ATM connaît un engouement formidable, aussi bien du côté des opérateurs de télécommunication que du côté des grands constructeurs informatiques. De ce fait, un très grand nombre de projets ont vu le jour. Ils concernent l'interconnexion de réseau, le contrôle du trafic, et la définition des services supportés par ATM.

Pour l'interconnexion de réseau, deux grandes méthodes sont en cours de définition [All95]. La première dans l'ordre chronologique est IP sur ATM définie par l'IETF [Col95, Cav92]. Elle est basée sur l'encapsulation des trames IP dans les trames AAL 5. De nombreuses RFC (Request For Comment) définissent son fonctionnement [RFC 1483, RFC 1577, RFC 1626, RFC 1755].

D'un autre côté, l'ATM Forum définit le concept de LAN Emulation [LANE]. Ce mécanisme est basé sur l'émulation des réseaux locaux 802.3 et 802.5 au niveau MAC, au-dessus de l'AAL 5.

Ces deux mécanismes permettent une interconnexion au niveau logiciel. Mais de nombreux travaux s'intéressent aux éléments matériels nécessaires à l'interconnexion de réseaux ATM avec les réseaux existants. En effet, il n'est pas possible d'imposer un nouveau réseau si celui-ci ne peut pas communiquer avec ceux déjà installés. Ainsi, de nombreux routeurs Ethernet avec des liens ATM apparaissent.

Le contrôle des débits dans le réseau public ATM intéresse de nombreuses équipes de recherche. En effet, il assurera la viabilité du réseau. Différentes méthodes de contrôle de congestion ont été proposées : la fenêtre sautante, la fenêtre glissante, le Leaky Bucket, le contrôleur espaceur du CNET (cf. paragraphe B.1.11). Elles permettent de répondre au besoin des services à largeur de bande garantie (CBR : Constant Bit Rate et VBR : Variable Bit Rate). Mais aujourd'hui, de nouveaux types de trafic sont introduits. Ainsi, les services élastiques tels que l'ABR (Available Bit Rate) et l'UBR (Unspecified Bit Rate) permettent d'utiliser le débit disponible sur le réseau à un instant donné [Gui94]. Des études sont à mener pour évaluer le comportement de ces services et leur impact sur les autres services et les congestions.

Du point de vue des services supportés par ATM, la télé-conférence, la visiophonie, le télé-enseignement, le calcul informatique réparti sont souvent cités. Aujourd'hui, les opérateurs de télécommunication cherchent de nouveaux types de services qu'ils pourront offrir sur le réseau public.

Enfin, le déploiement des réseaux ATM à grandes distances fait l'objet de nombreuses expérimentations. En France, le CNET et France Télécom ont conduit successivement le projet Sonate avec Alcatel pour la réalisation d'un système de communication ATM pour réseaux d'entreprise, et le projet Bréhat avec pour objectif la réalisation et l'évaluation d'un réseau ATM brassé (commutation sur les VP seulement). Ces études ont conduit à la mise en place d'un réseau ATM pilote à la fin de 1994, afin d'évaluer avec des utilisateurs réels le comportement du réseau.

Au niveau européen, de nombreuses études ont été financées par la communauté européenne dans le cadre des projets RACE. D'autre part, 18 opérateurs européens ont mis en place un réseau ATM pilote installé dans 15 pays d'Europe. La transmission s'effectue sur des liens PDH (Plesiochronous Digital Hierarchy) à 34 Mbit/s. Cette infrastructure a pour but de vérifier et valider les solutions retenues pour ATM, d'évaluer l'intérêt de différents services auprès d'utilisateurs pilotes, et d'expérimenter des services plus complexes entre centres de recherche. En Italie, une plate-forme ATM de taille importante a aussi été construite [Dec94].

Au niveau international, les projets ne manquent pas non plus. Le Japon s'intéresse au déploiement d'un réseau public basé directement sur la transmission ATM, sans passer par une autre technique de transfert comme SONET ou SDH. Aux Etats-Unis, de nombreux grands projets ont été financés par l'ARPA (Advanced Research Project Agency) et la NSF (National Science Foundation). D'autres sont issus de grandes sociétés.

Le programme XUNET (eXperimental University NETwork) a été mené à partir de 1986 en collaboration entre AT&T, l'Université de Californie à Berkeley, l'Université d'Illinois et l'Université du Wisconsin. A partir de 1990, il a fusionné avec le projet Blanca. La plate-forme de tests gigabit permet de mener des études sur les protocoles, l'interface entre les ordinateurs et les réseaux, les applications.

Le second projet de construction d'un réseau ATM de grande dimension est Aurora [Bie92, Cla92]. Il regroupe Bellcore, IBM, le MIT, l'Université de Pennsylvanie, MCI et NYMEX en reliant les quatre premiers d'entre eux par des liens SONET à 622 Mbit/s. Il s'intéresse à de nombreux aspects comme le contrôle de flux et la congestion [Cls92], l'architecture des protocoles [Cla90] et les interfaces [Lal92, Dav93, Tra93].

Le projet Nectar a pour but d'expérimenter des applications gigabit en réseau métropolitain. Il regroupe l'Université Carnegie-Mellon, Pittsburg Supercomputing Center, Bellcore et Bell Atlantic. Les deux premiers centres sont reliés par un lien SONET à 2,488 Gbit/s. Les sites ont des réseaux HIPPI et des réseaux ATM/SONET. Le projet s'intéresse particulièrement au calcul distribué [Joh92].

Le projet Zeus [Cox93] a pour but de construire un réseau ATM à l'intérieur de l'Université de Washington, qui supportera les services vidéo et de données. Il s'intéresse aux problèmes de congestion.

Le projet Magic [Ewy94] est l'une des plus récentes plate-forme ATM/SONET. Il regroupe un très grand nombre de participants et s'intéresse à deux aspects : les applications de visualisation sur les réseaux haut débit dans le cadre militaire, et l'intégration des protocoles tels que TCP/IP dans les réseaux ATM.

De nombreux autres projets de recherche sur les réseaux ATM, moins importants que ceux cités précédemment, existent aux Etats-Unis. Une description sommaire de plusieurs d'entre eux peut être trouvée dans [Par94].

En conclusion, la technologie ATM, et plus généralement le RNIS Large Bande, nécessite encore de nombreuses études avant un déploiement à grande échelle. Si son développement en réseau local est plus avancé qu'en réseau public, il reste des problèmes d'interconnexion, et surtout de contrôle du trafic.

B.4. Les recommandations CCITT pour le RNIS à Large Bande

- F.811 *Service support à large bande orienté connexion - 4 Août 1992*
- F.812 *Service support à large bande sans connexion pour données - 4 Août 1992*
- G.707 *Débits binaires de la hiérarchie numérique synchrone - 12 Mars 1993*
- G.708 *Interface de nœud de réseau pour la hiérarchie numérique synchrone - 12 Mars 1993*
- G.709 *Structure de multiplexage synchrone - 12 Mars 1993*
- I.113 *Glossaire des termes relatifs au RNIS Large Bande - 5 Avril 1991*
- I.121 *Aspects Large Bande du RNIS - 5 Avril 1991*
- I.150 *Caractéristiques fonctionnelles du mode de transfert asynchrone du RNIS à Large Bande - 12 Mars 1993*
- I.211 *Aspects service du RNIS à Large Bande - 12 Mars 1993*
- I.311 *Aspects généraux du réseau pour le RNIS à Large Bande - 12 Mars 1993*
- I.321 *Modèle de référence pour le protocole du RNIS à Large Bande et son application - 5 Avril 1991*
- I.327 *Architecture fonctionnelle du RNIS à Large Bande - 12 Mars 1993*
- I.356 *Performance du transfert de cellules dans la couche mode de transfert asynchrone du RNIS à Large Bande - 26 Novembre 1993*
- I.361 *Spécifications de la couche Mode de Transfert Asynchrone pour le RNIS à Large Bande - 12 Mars 1993 - Etudes complémentaires sur l'utilisation du champ GFC de l'en-tête de la cellule ATM*
- I.362 *Description fonctionnelle de la couche Adaptation du mode de transfert asynchrone (AAL) du RNIS à Large Bande - 12 Mars 1993*
- I.363 *Spécification de la couche d'adaptation ATM du RNIS Large Bande - 12 Mars 1993 - Discussions en cours, principalement pour l'AAL 2*
- I.364 *Fourniture d'un service de transmission de données sans connexion sur le RNIS à Large Bande - 12 Mars 1993 - Définition en cours*
- I.371 *Gestion du trafic et des encombrements dans le RNIS à Large Bande - 12 Mars 1993 - Définition en cours*
- I.413 *Interface usager-réseau du RNIS à Large Bande - 12 Mars 1993*

- I.414 *Vue d'ensemble des recommandations relatives à la couche 1 pour l'accès d'abonné au RNIS et au RNIS à Large Bande - 12 Mars 1993*
- I.432 *Interface usager-réseau du RNIS à Large Bande - Spécification de la couche physique - 12 Mars 1993*
- I.610 *Principes et fonctions d'exploitation et de maintenance du RNIS à Large Bande - 12 Mars 1993 - Discussions en cours*
- Q.2100 *B-ISDN Signalling ATM Adaptation Layer Overview Description - Décembre 1993 - (précédemment Q.SAAL 0) - Définition en cours*
- Q.2110 *B-ISDN - ATM Adaptation Layer - Service Specific Connection Oriented Protocol (SSCOP) - Décembre 1993 - (précédemment Q.SAAL 1) - Définition en cours*
- Q.2130 *B-ISDN Signalling ATM Adaptation Layer - Service Specific Coordination Function for Support of Signalling at the User-to-Network Interface (SSCF at UNI) - Décembre 1993 - (précédemment Q.SAAL 2) - Définition en cours*
- Q.2140 *B-ISDN Signalling ATM Adaptation Layer - Service Specific Coordination Function for Support of Signalling at the Network Node Interface (SSCF at NNI) - Définition en cours*
- Q.2931 *Broadband Integrated Services Digital Network (B-ISDN) Digital Subscriber Signalling System No. 2 (DSS 2) User Network Interface Layer 3 Specification for Basic Call/Connection Control - Draft - Juin 1994 - (précédemment Q.93b) - Définition en cours*

B.5. Les spécifications de l'ATM Forum

- B-ICI *A draft of the B-ICI Specification Document version 2.0 - Février 1995*
- IISP *Interim Inter-Switch Signaling Protocol - Février 1995*
- LANE *LAN Emulation Over ATM: Version 1.0 Specification - 6 Janvier 1995*
- UNI 3.1 *ATM User-Network Interface Specification version 3.1 - Janvier 1995*

B.6. Les documents de l'IETF

- RFC 1483 *Multiprotocol encapsulation over ATM Adaptation Layer 5 - J. Heinanen - Juillet 1993*

- RFC 1577 *Classical IP and ARP over ATM* - M. Laubach - Janvier 1994
- RFC 1595 *Definitions of Managed Objects for the SONET/SDH Interface Type* - T. Brown, K. Tesink - 11 Mars 1994
- RFC 1626 *Default IP MTU for use over ATM AAL5* - R.J. Atkinson - Mai 1994
- RFC 1633 *Integrated Services in the Internet Architecture: an Overview* - R. Braden, D. Clark, S. Shenker - Juin 1994
- RFC 1695 *Definitions of Managed Objects for ATM Management Version 8.0 using SMIv2* - M. Ahmed, K. Tesink - 25 Août 1994
- RFC 1754 *IP over ATM Working Group's Recommendations for the ATM Forum's Multiprotocol BOF Version 1* - M. Laubach - Janvier 1995
- RFC 1755 *ATM Signalling Support for IP over ATM* - M. Perez, F. Liaw, A. Mankin, E. Hoffman, D. Grossman, A. Malis - Février 1995

B.7. Références relatives à l'annexe B

- [All95] A. Alles - *"ATM Internetworking"* - Rapport technique - Cisco Systems - Mars 1995
- [Ans93] *"A Technical Report on a comparison of SONET and SDH"* - rapport ANSI T1X1.2/93-024R2
- [Bel91] *"Report on B-ISDN protocols for providing SMDS and exchange access SMDS"* - Rapport de recherche - Bellcore - Septembre 1991
- [Bie92] E.W. Biersack, C.J. Cotton, D.C. Feldmeier, A.J. McAuley, W.D. Sincoskie - *"Gigabit Networking Research at Bellcore"* - IEEE Network - Mars 1992
- [Bou92] J.Y. Le Boudec - *"The Asynchronous Transfer Mode : a tutorial"* - Computer Networks and ISDN Systems / North Holland - vol. 24 - n° 4 - 15 Mai 1992 - p 279-309
- [Boy92] P.E. Boyer, D.P. Tranchier - *"A reservation principle with applications to the ATM traffic control"* - Computer Networks and ISDN Systems - vol. 24 - n° 4 - 15 Mai 1992 - p 321-334
- [Cas92] A.J. Casaca, M.S. Nunes - *"Some aspects of the adaptation of services to an ATM network"* - Proc. of 4th Information Network and Data Communication - Espoo - Finlande - 16-19 Mars 1992 - p 107-120
- [Cav92] J.D. Cavanaugh, T.J. Salo - *"Internetworking with ATM WANs"* - Rapport - Minnesota Supercomputer Center - 14 Décembre 1992

- [Cla90] D.D. Clark, D.L. Tennenhouse - *"Architectural Considerations for a New Generation of Protocols"* - Proceedings of ACM SIGCOMM'90 - Septembre 1990 - p 200-208
- [Cla92] D.D. Clark, B.S. Davie, D.J. Farber, I.S. Gopal, B.K. Kabada, W.D. Sincoskie, J.M. Smith, D.L. Tennenhouse - *"An Overview of the AURORA Gigabit Testbed"* - Proc. of IEEE INFOCOM 1992 - Florence - Italie - 1992
- [Cls92] D.D. Clark, S. Shenker, L. Zhang - *"Supporting real-time applications in an integrated services packet network : architecture and mechanism"* - Proceedings of ACM SIGCOMM'92 - Août 1992 - p 14-26
- [Cne91] L'écho des recherches - *"Spécial ATM"* - Publications CNET - n° 144 & 145 - 1991
- [Col95] R.G. Cole, D. Shur - *"IP Over ATM: A Framework Document"* - Internet Draft - AT&T Bell Laboratories - 27 Juin 1995
- [Cox93] J.R. Cox, M.E. Gaddis, J.S. Turner - *"Zeus Project"* - IEEE Network - Mars 1993 - vol. 7 - n° 2 - p 20-30
- [Dav93] B.S. Davie - *"The Architecture and Implementation of a High-Speed Host Interface"* - IEEE Journal on Selected Areas in Communications - vol. 11 - n° 2 - Février 1993 - p 228-239
- [Dec94] M. Decina, C. Mossotto, A. Roveri - *"The ATM Test-Bed: An Experimental Platform for Broadband Communications"* - IEEE Communications Magazine - Octobre 1994 - p 78-83
- [Ewy94] B.J. Ewy, J.B. Evans, V.S. Frost, G.J. Minden - *"TCP/ATM Experiences in the MAGIC Testbed"* - Gigabit Network Workshop 94 - 12 Juin 1994
- [Gui92] F. Guillemin, A. Dupuis - *"A basic requirement for the policing function in ATM networks"* - Computer Networks and ISDN Systems - vol. 24 - n° 4 - 15 Mai 1992 - p 311-320
- [Gui94] F. Guillemin, A. Dupuis - *"Les services élastiques sur l'ATM"* - Rapport CNET NT/LAA/RSL/17 - Septembre 1994
- [Gul92] F. Guillemin, - *"Contrôle de congestion dans les réseaux temporels asynchrones"* - Thèse à l'Université de Rennes 1 - 3 Juillet 1992
- [Joh92] C.A. Johnston, K.C. Young, K.A. Walsh, N.K. Cheung - *"A programmable ATM/AAL interface for gigabit network applications"* - Globecom'92 - Décembre 1992 - vol. 2 - p 1191-1195
- [Kun92] A.T. Kundig - *"ISDN and B-ISDN : from basic technology to applications"* - Proc. of 4th Information Network and Data Communication - Espoo - Finlande - 16-19 Mars 1992 - p 83-90

- [Lal92] G.R. Lalk, B.S. Davie, K.C. Young - *"An OC-12/STS-3c/ATM interface for gigabit network applications"* - Electronics Letters - vol. 28 - n° 24 - 19 Novembre 1992 - p 2204-2206
- [Lem93] M. Lemerancier, G. Pujolle - *"Contrôle d'accès dans les réseaux ATM, études des performances des "leaky buckets" "* - Réseaux et informatique répartie - vol. 3 - n° 3 - 1993 - p 267-286
- [Lim88] Y. Lim, J. Kobza - *"Analysis of Delay Dependent Priority Discipline in a Multiclass Traffic Packet Switching Node"* - INFOCOM - 1988 - p 889-898
- [Lyo91] T. Lyon - *"Simple and efficient layer (SEAL)"* - Rapport - Sun Microsystems Inc. - Août 1991
- [Par94] C. Partridge - *"Les réseaux gigabits"* - Addison-Wesley - Août 1994
- [Pry91] M. De Prycker, R. Peshi, T. Van Landegem - *"B-ISDN and the OSI protocol reference model"* - Proceedings of the 3rd Conference on High Speed Networking - Berlin - 18-22 Mars 1991 - p 39-57
- [Rat91] E.P. Rathgeb - *"Modelling and Performance Comparison of Policing Mechanisms for ATM Networks"* - Journal on Selected Areas in Communications - vol. 9 - n° 3 - Avril 1991
- [Sai90] H. Saito - *"Optimal Queueing Discipline for Real-Time Traffic at ATM Switching Nodes"* - IEEE Transactions on Communications - vol. 38 - n° 12 - 1990 - p 2131-2136
- [Sch90] H. Schneider - *"The concept of virtual paths and virtual channels in ATM networks"* - Proc. of the International Seminar on Digital Communications - Zurich - 1990 - p 63-72
- [Sid89] M. Sidi, W.Z. Liu, I. Cidon, I. Gopal - *"Congestion Control through Input Rate Regulation"* - Globecom - Dallas - Texas - 1989
- [Tra93] C.B.S. Traw, J.M. Smith - *"Hardware/software organization of a high performance ATM host interface"* - IEEE Journal on Selected Areas in Communications - vol. 11 - n° 2 - Février 1993 - p 240-253
- [Tur86] J. Turner - *"New directions in communications (or which way in the information age?)"* - IEEE Communications Magazine - vol. 24 - Octobre 1986 - p 8-15

Annexe C

La carte ATM développée

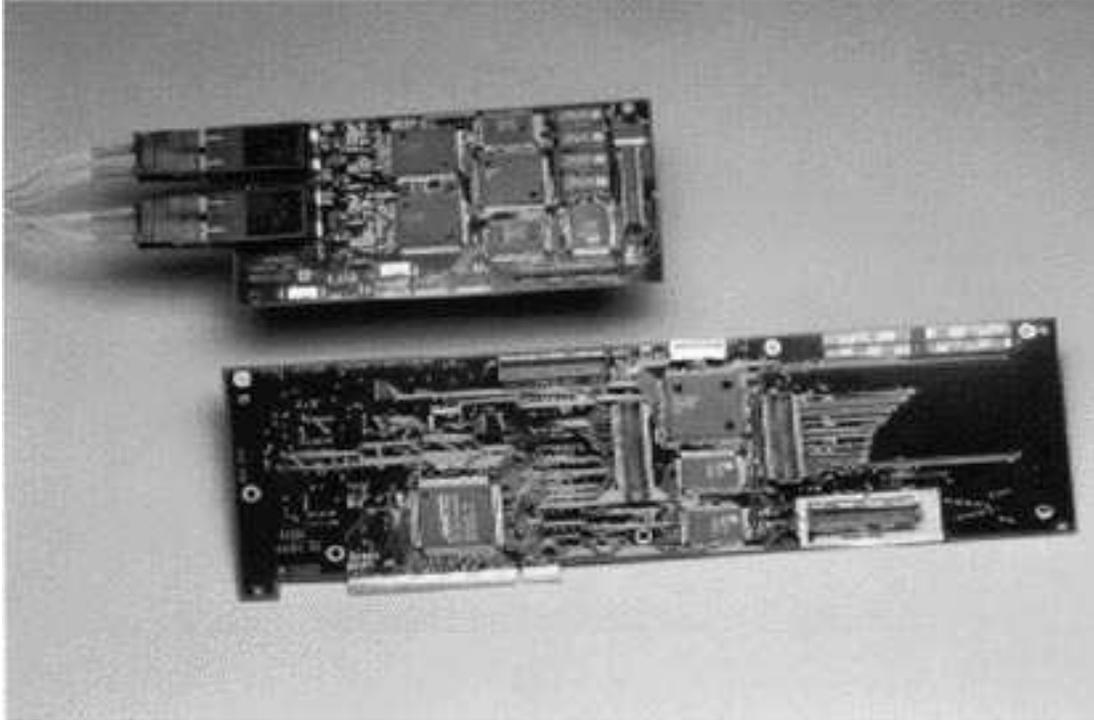


Figure C.1 : La carte mère d'adaptation à ATM et sa carte fille d'interface physique

Résumé

L'évolution des réseaux de communication rend aujourd'hui possible l'apparition de nouvelles applications distribuées. Néanmoins, l'émergence du multimédia renforce les besoins de bande passante élevée et de garantie des délais, qui devront être pris en considération tant au niveau du réseau que des machines.

Cette thèse propose l'architecture d'un système de communication complet, basé sur ATM, permettant de satisfaire les besoins des applications distribuées temps réel. En particulier, il est nécessaire d'intégrer de manière efficace les flux audio et vidéo, qui devront cohabiter avec d'autres trafics comme les alarmes ou les données traditionnelles.

L'état de l'art sur l'architecture des machines pour les hauts débits permet de mieux cerner les problèmes posés par l'emploi des nouvelles technologies de communication, et montre que les nombreuses études dans ce domaine ne fournissent que des réponses partielles au problème posé.

L'architecture du système de communication proposée permet de supporter plusieurs équipements audio/vidéo sur une même station, de construire des topologies de réseau variées, et d'assurer la redondance de chemins. L'introduction d'un micro-contrôleur sur la carte d'interface au réseau offre des services temps réel de haut niveau grâce au concept de télé-DMA, et fait ainsi profiter les applications des avantages d'ATM. D'autre part, une organisation protocolaire particulière permet de faire cohabiter efficacement les services temps réel et les services télé-informatique classiques.

Pour les communications entre les équipements, un mécanisme de contrôle de flux à la source pour garantir un délai d'acheminement borné des messages temps réel est proposé. Les problèmes de récupération des applications existantes, de signalisation, d'administration et de gestion du basculement en cas de défaillance du réseau sont aussi abordés.

Enfin, les performances globales du système de communication, puis celles d'un service de transfert périodique de variables temps réel implanté sur le micro-contrôleur sont évaluées.

Mots-clés : Système de communication, Protocoles haut débit, ATM, Applications distribuées, Temps réel, Multimédia, Interface réseau, Architecture, Performances, VHDL.

Abstract

The progress in communication networks enables the deployment of new distributed applications. Nevertheless, large bandwidth and delay guaranty requirements grow with multimedia traffic. These constraints should be taken into consideration at the network as well as the workstation levels.

This thesis proposes the architecture of an entire communication system based on ATM that satisfies the real-time distributed application requirements. In particular, audio and video streams must be integrated efficiently, and must operate with other traffics such as alarms and traditional data.

The state of the art on high speed machine architecture allows to better determine the problems raised by the use of new communication technologies. It presents numerous studies in this field and shows that they only gives partial solutions to the problem.

The proposed communication system architecture supports several audio/video devices on a unique workstation, allows to build various network topologies, and provides path redundancy. The addition of a micro-controller on the network interface board offers high-level real-time services thanks to the tele-DMA concept. Thus, applications can take advantage of the ATM technology. Furthermore, a special protocol organization allows to support in an efficient way real-time services as well as traditional distributed applications.

Concerning the inter-device communication, a source flow control is proposed in order to guaranty a bounded transmission delay for real-time messages. The re-use of existing application, the signaling, the network management and the recovery process when the network fails are discussed.

Finally, the communication system general performances, and the periodic real-time update of control variables implemented on the micro-controller are evaluated.

Keywords : Communication system, High speed protocols, ATM, Distributed applications, Real-time, Multimedia, Network interface, Architecture, Performances, VHDL.