Mémoire d'Habilitation à Diriger des Recherches

Institut National Polytechnique de Toulouse

* * *

CONTRIBUTION AUX PROTOCOLES ET AUX ARCHITECTURES DE COMMUNICATION DE BOUT EN BOUT POUR LA QDS DANS L'INTERNET

Christophe CHASSOT

Maître de Conférences à l'INSA de Toulouse
Chercheur au LAAS-CNRS

* * *

Soutenue au LAAS-CNRS le 12 décembre 2005 devant le jury :

Serge FDIDA, Professeur à l'Université P. et M. Curie, Paris	Rapporteur
Francis LEPAGE, Professeur à l'Université H. Poincaré, Nancy	Rapporteur
Michel DIAZ, Directeur de Recherche au LAAS-CNRS	Correspondant
Jean Marie DILHAC, Professeur à l'INSA de Toulouse	Examinateur
Christian FRABOUL, Professeur à l'ENSEEIHT, Toulouse	Examinateur
Guy JUANOLE, Professeur à l'Université P. Sabatier, Toulouse	Examinateur

Guide de lecture

Ce mémoire est composé de deux parties distinctes.

La première partie résume les activités d'enseignement et de recherche que j'ai eues depuis mon entrée au LAAS-CNRS en 1992 et à l'INSA de Toulouse en 1996 :

- sur la partie enseignement, elle décrit les responsabilités pédagogiques et administratives qui m'ont été confiées ;
- sur la partie recherche, elle présente la liste des publications, les encadrements, les collaborations industrielles et de recherche, ainsi que les séminaires invités et les participations à jury de thèse.

La deuxième partie présente les travaux de recherche que j'ai menés au LAAS-CNRS de 1992 à aujourd'hui, ainsi que ma prospective de recherche. Elle est structurée en cinq parties :

- un état de l'art des thèmes de recherche abordés ;
- les trois étapes des travaux de recherche effectués ;
- la prospective pour les années à venir.

Tant en recherche qu'en enseignement, les travaux ici décrits ont été réalisés en collaboration avec de nombreux collègues, doctorants et stagiaires, ce qui explique dans la suite l'utilisation fréquente de la première personne du pluriel pour décrire *nos* contributions.

* * *

Résumé. Ces dernières années, les évolutions de l'informatique et des télécommunications ont conduit à une modification drastique du paysage de la communication informatique, et en conséquence de l'Internet et de ses services : les applications sont désormais à la fois multimédias, multi utilisateurs et coopératives ; les technologies réseaux offrent à présent de hauts débits de transmission, à échelle locale ou grande distance, et permettent de se connecter à l'Internet via différents types de terminaux et de points d'accès, filaires ou sans fil. En dépit de ces avancées, les services de communication offerts par l'Internet sont encore très insuffisants, en ceci qu'ils n'offrent aucune garantie sur les performances offertes, notamment en termes de délai de bout en bout. C'est dans cette problématique que s'inscrivent nos travaux, qui adressent le besoin de nouveaux services, protocoles et architectures de communication pour l'Internet, dans le but d'offrir des garanties de Qualité de Service (QdS). Notre démarche part d'une expression formelle des besoins applicatifs et aboutit à des solutions de bout en bout visant à : (1) intégrer les niveaux considérés (Application, Transport et IP) ; (2) minimiser l'utilisation des ressources (bande passante, buffer, …) ; et (3) abstraire le niveau applicatif de la complexité du choix et du paramétrage des services sous-jacents. Trois thèmes de recherche sont explorés : (1) les services et protocoles de Transport multimédia pour optimiser la QdS dans un contexte IP de type Best Effort ; (2) les architectures de bout en bout pour garantir la QdS, en coordonnant les nouveaux services de l'Internet aux niveaux Transport et IP (IntServ et DiffServ mono puis multi domaines) ; et (3) la signalisation pour la QdS, dans un contexte IP de type DiffServ multi domaines. Notre prospective concerne d'une part la signalisation pour la QdS en considérant à présent l'hétérogénéité de l'Internet multi domaines, et d'autre part, les protocoles de Transport auto configurables et les architectures dynamiques, pour optimiser la qualité des communications et des coopérations dans les futurs réseaux *ambiants* (réseaux mobiles et ad hoc, capteurs, …) en tenant compte d'une dynamicité du contexte à la fois au niveau utilisateur, mobile et coopérant, et au niveau du réseau, mobile et aux ressources variables.

Mots clefs. Internet multi domaines, Qualité de service, Architecture, Service et protocole de Transport, IntServ, DiffServ, Signalisation, Internet ambiant, Auto adaptabilité, Protocole auto configurable, Architecture dynamique.

Summary. These latest years, the evolutions of computer science and telecommunications led to a radical modification of the communication landscape, and consequently of the Internet and its services: applications are now multimedia, multi users and co-operatives; networks now provide high transmission rates, at local scale or long distance, and make it possible to be connected to the Internet via various types of terminals and wire/wireless access points. In spite of these advances, the Internet communication services are still insufficient, in this way that they provide no guarantee on the performances, particularly as far as the end-to-end transit delay is concerned. This problem is targeted by our work, which addresses the need of new services, protocols and communication architectures for the Internet, aimed at providing end-to-end Quality of Service (QdS) guarantees. We start from a formal expression of the application needs, and lead to end-end solutions which try: (1) to integrate the considered levels (Application, Transport and IP); (2) to minimize resources utilization (band-width, buffer...); and (3) to abstract the application level from the choice and the configuration of the underlying services. Our contributions address three research topics: (1) the multimedia Transport services and protocols aimed at optimizing QdS in the Best Effort Internet; (2) the architectures providing guaranteed QdS, by coordinating the new services of the Internet at both Transport and IP levels (IntServ and DiffServ mono then multi domains); and (3) the signalling for QdS in a multi domains DiffServ Internet. Our research prospective concerns: (1) the signaling for QoS considering the *heterogeneity* of the multi domains Internet, and (2) the dynamic architecture and the configurable protocols to optimize the quality of the communication and the cooperation in the future ambient networks (mobile and ad hoc networks, sensors...) by taking into account the dynamicity of the context both at the user level (mobile and cooperating), and at the network level (mobile and with variable resources).

Key words. Multi domain Internet, Quality of service, Architecture, Transport service and protocol, IntServ, DiffServ, Signaling, Ambient Internet, Context awareness, Adaptability, Self configurable protocol, Dynamic architecture

LISTE DES ACRONYMES

@IRS	Architecture Intégrée de Réseaux et Services
AEI	Automatique et Electronique Intégrée
AF	Assured Forwarding
API	Application Programming Interface
AQUILA	Adaptive Resource Control for QoS Using an IP-based Layered Architecture
AS	Assured Service
BART_L(N)	Buffer, Acknowledᵗ, Retransmission, Timer - Losable after N retransmissions
BART_NL	Buffer, Acknowledgement, Retransmission, Time - Not Losable
BGP	Border Gateway Protocol
BGRP	Border Gateway Reservation Protocol
CAC	Connection Admission Control
CADENUS	Creation and deployment of end-user services in premium IP networks
CBR	Constant Bit Rate
CESAME	Conception formElle de Systèmes hAuts débits Multimédia coopEratifs
CFIP	Colloque francophone sur l'ingénierie des protocoles.
CL	Controlled Load
COPS	Common Open Policy Service protocol
COPS-PR	COPS Provisionning.
DCCP	Datagram Congestion Control Protocol
DGA	Direction Générale de l'Armement
DiffServ	Differentiated Service Working Group
DIS	Distributed Interactive Simulation
EDIT	Ecole Doctorale Informatique et Télécommunication, Toulouse
EDSYS	Ecole Doctorale Système, Toulouse
EF	Expedited Forwarding
ENSEEIHT	Ecole Nationale Sup. d'Electronique, d'Electrotechnique, d'Informatique, d'Hydraulique et de Télécommunication de Toulouse
EuQoS	End to end QoS support over heterogeneous networks
FPTP	Fully Programmable Transport Protocol
GEI	Génie Electrique et Informatique
GI	Génie Informatique
GII	Génie Informatique Industriel
GIMPS	General Internet Messaging Protocol for Signaling
GIST	General Internet Signaling Transport protocol
GS	Guaranteed Service
IHM	Interface Homme Machine
IMACS	Ingénierie des Matériaux, Composants et Systèmes
INPT	Institut National Polytechnique de Toulouse
INSA	Institut National des Sciences Appliquées
IntServ	Integrated Service Working Group
LAAS	Laboratoire d'Analyse et d'Architecture des Systèmes
LESIA	Laboratoire d'Etude des Systèmes Informatiques et Automatiques
MESCAL	Management of End-to-end Quality of Service Across the Internet at Large.
MESR	Ministère Enseignement Supérieur et de la Recherche
MIC	Modélisation, Informatique et Communication
MM-POC	Multimedia Partial Order Connection
NBART_L	No Buffer, Acknowledgement, Retransmission - Losable
NETBLT	NETwork BLock
NGNI	Next Generation Networks Initiative

NSIS	Next Step In Signaling Protocol
NSLP	NSIS Signaling Layer Protocol
NTD	Network Technology Dependant
NTI	Network Technology Independent
NTLP	NSIS Transport Layer Protocol
OLC	Outils et Logiciels pour la Communication
PBN	Policy based Networking
PBS	Partial Buffer Sharing
PDU	Protocol Data Unit
PEDR	Prime d'Encadrement Doctoral et de Recherche
PDP	Policy Decision Point
PEP	Policy Enforcement Point
PHB	Per Hop Behaviour
POC	Partial Order Connection
QBone	QoS Bone
RAA	Ressource Allocation Answer
RAR	Ressource Allocation Request
RM	Resource Manager
RMAPI	Resource Manager API
RNRT	Réseau National de la Recherche en Télécommunications
RSVP	ReSource reserVation Protocol
RT	Réseaux et Télécom
RTSA	Réseaux, Télécom, Système et Architecture
SIBBS	Simple Inter domain Bandwidth Broker Signalling protocol.
SLA	Service Level Agreement
SLS	Service Level Specification.
TCP	Transmission Control Protocol
TEQUILA	Traffic Engineering for Quality of Service in the Internet, at Large Scale
TSPEC	Traffic Specification
TSPN	Timed Stream Petri Nets
UDP	User Datagram Protocol
VMTP	Versatile Message Transaction Protocol
VP	Virtual Path
XTP	Xpress Transfer Protcol

* * *

Table des matières

PARTIE 1 : ACTIVITES DE RECHERCHE ET D'ENSEIGNEMENT

I. Renseignements personnels 15
1. Etat civil 15
2. Diplômes 15
3. Cursus 15
II. Activités de recherche 16
1. Publications 16
1.1. Résumé 16
1.2. Liste détaillée (hors rapports de contrat) 16
2. Encadrements 19
2.1. Résumé 19
2.2. Encadrements de thèse 19
2.3. Encadrements de stage de DEA/MASTER Recherche (et de fin d'étude ingénieur) 20
3. Animation scientifique 20
3.1. Collaborations industrielles 20
3.2. Thème SCD de la fédération inter laboratoires FERIA 23
3.3. Séminaires invités 23
3.4. Participation à jury de thèse 24
III. Activités d'enseignement 24
1. Introduction 24
2. Enseignements 26
2.1. Résumé 26
2.2. Détails des enseignements 27
3. Elaboration de nouveaux programmes d'enseignement 29
3.1. Orientation de 5e année Réseaux et Télécommunications 29
3.2. Spécialité Réseaux et Télécommunications 30

PARTIE 2 : TRAVAUX DE RECHERCHE

INTRODUCTION 35
ETAT DE L'ART 39
I. Les services et protocoles de Transport à QdS 39
1. Introduction 39
2. Problématique 39
2.1. Evolution du contexte applicatif et réseau 39
2.2. Limites des solutions initiales 40
3. Propositions 41
3.1. Amélioration des mécanismes à des fins de performances 41
3.2. Extensions conceptuelles 43
4. Conclusions 45

II. Les architectures pour la QdS de bout en bout dans l'Internet 46
1. Introduction 46
1.1. Deux modèles précurseurs : IntServ et DiffServ 46
1.2. Problématique générale 48
2. Architectures à QdS basées sur IntServ 49
2.1. Problèmes 49
2.2. Propositions 50
2.3. Conclusions 51
3. Architectures à QdS basées sur DiffServ 52
3.1. Problèmes 52
3.2. Propositions 54
3.3. Conclusions 62
III. La signalisation pour la QdS dans l'Internet multi domaines 63
1. Introduction 63
2. Problèmes 64
2.1. Vis à vis du provisionnement 64
2.2. Vis à vis de la disponibilité des ressources 65
2.3. Vis à vis des modèles à base de BB 65
3. Propositions 66
3.1. L'approche SCCD (signalisation couplée au chemin de donnée) 66
3.2. L'approche SDCD (signalisation découplée du chemin de donnée) 67
4. Conclusion 70

ETAPE 1 : CONTRIBUTION AUX SERVICES ET PROTOCOLES DE TRANSPORT A QdS 73

I. Introduction 73
II. POC : une famille de nouveaux protocoles 73
1. Le concept de connexion d'ordre partiel et de fiabilité partielle 73
2. POC : un support adapté au transfert des flux multimédia 74
2.1. Intérêt d'un Transport à fiabilité partielle 74
2.2. Intérêt d'un Transport à ordre partiel 75
3. Proposition d'architecture pour un Transport multimédia 75
3.1. Principes de conception 76
3.2. Impact d'une gestion intégrée de l'ordre et de la fiabilité 77
III. Evaluation des performances 78
IV. Conclusion 80

ETAPE 2 : CONTRIBUTION AUX ARCHITECTURES A QdS EN CONTEXTE INTSERV 83

I. Introduction 83
II. Caractérisation de la QdS requise par les applications DIS 83
1. Introduction à la simulation interactive distribuée 83
1.1. Généralités 83
1.2. Caractéristiques du protocole DIS 84
2. QdS requise par une application DIS 85
2.1. Limites du standard DIS 85
2.2. Proposition de QdS 87
III. Caractérisation du trafic DIS 90
1. Principe 90

2. Validation expérimentale 91
IV. Architecture DIS/ATM 91
1. Principes 92
1.1. DIS+ 92
1.2. RSVM 93
2. Déploiement et mesures de performance 93
V. Conclusion 94

ETAPE 3 : CONTRIBUTION AUX ARCHITECTURES A QdS EN CONTEXTE DIFFSERV 97

I. Introduction 97
II. Architecture @IRS 97
1. Choix d'architecture 98
1.1. Principes généraux 98
1.2. Fonctions des plans de contrôle et de données du niveau IP 99
2. Implémentation et validation 99
2.1. Plateforme @IRS 100
2.2. Mesures de performance 100
III. Architecture @IRS++ 102
1. Extension des expérimentations @IRS par simulation (ns-2) 102
1.1. Topologie étudiées en simulation 103
1.2. Résultats et conclusions 104
2. Le modèle de performance 104
3. API et algorithme de sélection 105
3.1. API 105
3.2. Algorithme de sélection 106
IV. Architecture @IRS++/MD 108
1. Modèle de performance d'un canal en contexte multi domaines 109
1.1. Modèle mathématique sous-jacent à l'étude 110
1.2. Application à l'étude 110
2. Architecture 111
2.1. Approche adoptée pour @IRS++/MD 111
2.2. Signalisation 112
V. Conclusion 117

CONCLUSION ET PROSPECTIVE 119

I. Conclusion générale 119
II. Prospective 120
1. Signalisation pour l'Internet multi domaines hétérogènes 120
2. Protocoles de Transport auto configurables et architectures auto adaptatives 122

REFERENCES BIBLIOGRAPHIQUES 127

Partie 1

Activités de Recherche et d'Enseignement

I. Renseignements personnels

1. Etat civil

Christophe Chassot

- Né le 21 octobre 1966 à Lyon (Rhône), nationalité française
- 17 rue de Chateaudun, 31000 Toulouse
- Maître de Conférences à l'INSA[1] de Toulouse, Département GEI[2]
- Travaux de recherche effectués au LAAS[3] du CNRS, groupe OLC[4]
- Email et téléphones professionnels :
  - INSA : chassot@insa-toulouse.fr / 05 61 58 98 36
  - LAAS : chassot@laas.fr / 05 61 33 78 16

2. Diplômes

- **Juin 1992** : Diplôme d'Ingénieur en Informatique et Math. Appliquées de l'ENSEEIHT[5]
- **Septembre 1992** : DEA Informatique Fondamentale et Parallélisme de l'INPT[6]
- **Décembre 1995** : Doctorat de l'INPT, Spécialité Informatique

3. Cursus

- **1985/1988** : Classes préparatoires au lycée Paul Cézanne d'Aix en Provence
- **1988/1989** : Service militaire effectué à Nancy (Escadron d'Instruction)
- **1989/1992** : Elève ingénieur ENSEEIHT spécialité Informatique et Math. Appliquées
- **1992/1995** : Formation doctorale au LAAS (OLC). Directeur : Michel Diaz, DR au CNRS
- **1995/1996** : ATER (poste plein) à l'INSA de Toulouse (DGEI)
- **depuis septembre 1996** :
  - Maître de Conférences à l'INSA de Toulouse, DGEI (6ème échelon) - 61ème section CNU
  - Spécialité Réseaux et Informatique
  - Activités de recherche effectuées au sein du groupe OLC du LAAS-CNRS
  - Titularisation en septembre 1997
  - Passage à l'ex grade de 1ère classe en septembre 2000
  - PEDR[7] de 2000 à 2003

---

[1] INSA : Institut National des Sciences Appliquées

[2] GEI : Génie Electrique et Informatique

[3] LAAS : Laboratoire d'Analyse et d'Architecture des Systèmes

[4] OLC : Outils et Logiciels pour la Communication

[5] ENSEEIHT : Ecole Nationale Sup. d'Electronique, d'Electrotech., d'Informatique, d'Hydraulique et de Télécom. de Toulouse

[6] INPT : Institut National Polytechnique de Toulouse

[7] PEDR : Prime d'Encadrement Doctoral et de Recherche

II. Activités de recherche

Cette section résume nos activités de recherche en termes de publications, d'encadrements et d'animation scientifique. Nos thématiques de recherche et nos contributions sont détaillées dans la deuxième partie de ce mémoire.

1. Publications

1.1. Résumé

- Thèse : 1
- Revues internationales : 5
- Revues nationales : 4
- Participation à ouvrage : 2
- Conférences internationales avec acte et comité de lecture : 15
- Conférences francophones[8] avec acte et comité de lecture : 6
- Conférences nationales avec acte et comité de lecture : 3
- Rapports de contrat : 12

1.2. Liste détaillée (hors rapports de contrat)

Thèse

1. Chassot C. Architecture de Transport Multimédia à Connexions d'Ordre Partiel. Thèse de Doctorat en Informatique de l'Institut National Polytechnique de Toulouse (INPT), Décembre 1995.

Revues internationales : 5

2. Thai K-L, Chassot C, Fdida S, Diaz M. Transport Layer for Cooperative Multimedia Application. Annals of Telecommunications, May-June 94, vol. 49, n°5-6.
3. Diaz M, Lozes A, Chassot C, Amer P. Partial Order Connections. A new concept for High Speed and Multimedia Services and Protocols. Annals of Telecommunications, May-June 94, vol. 49, n°5-6.
4. Amer P, Chassot C, Connolly C, Conrad P, Diaz M. Partial Order Transport Service for Multimedia and other Applications. IEEE/ACM Transactions on Networking, October 1994, vol. 2, n° 5.
5. Chassot C, Diaz M, Lozes A. From the Partial Order Concept to Partial Order Multimedia Connections. Journal for High Speed Networks, vol.5, n°2, 1996.
6. Owezarski P, Chassot C, Diaz M. A Time Efficient Architecture For Multimedia Applications. IEEE Journal on Selected Areas in Communication (JSAC), April 1998, vol.16, n°3.

Revues nationales : 4

7. Chassot C, Lozes A, Fournier M, Diaz M. Evaluation des performances d'une architecture de transport multimédia. Technique et Science Informatiques (TSI), Décembre 1999.

[8] CFIP : Colloque francophone sur l'ingénierie des protocoles.

8. Chassot C, Lozes A, Garcia F, Diaz M, Dairaine L, Rojas L. Caractérisation et mise en œuvre de la QdS requise par une application DIS distribuée dans un environnement Internet de nouvelle génération. Revue Electronique sur les Réseaux et l'Informatique Répartie (RERIR), Avril 2000.
9. Lochin E, Anelli P, Fdida S, Auriol G, Garcia F, Chassot C, Lozes A. Evaluation de la différenciation de service dans l'Internet. Technique et Science Informatique (TSI), vol. 23, n°5-6, Décembre 2004.
10. Auriol G, Chassot C, Diaz M. Architecture de communication à gestion automatique de la QdS en environnement IP à services différenciés. Technique et Science Informatique (TSI), vol. 23, n°9, Décembre 2004.

Participation à ouvrage : 2

11. Owezarski P., Chassot C. La couche Transport et ses évolutions. Logiciels et réseaux de com. : compte rendu du groupe Logiciels et réseaux de communication de l'OFTA, ARAGO n° 23, Ed. Lavoisier, Mai 2000.
12. Boyer M, Chassot C, Dairaine D, Diaz M, Lozes A, Owezarski P, Rojas L. Protocoles de Transport multimédia. Systèmes multimédia communicants, Ed. W. Dabbous, Hermès, Juin 2001.

Conférences internationales avec acte et comité de lecture : 15

13. Amer P, Chassot C, Connolly C, Diaz M. Partial Order Transport Service for Multimedia Application - Reliable Service. 2nd High Performance Distributed Computing Conference, July 1993.
14. Amer P, Chassot C, Connolly C, Diaz M. Partial Order Transport Service for Multimedia Applications - Unreliable Service. 3rd International Networking Conference, INET'93, August 1993.
15. Diaz M, Drira K, Lozes A, Chassot C. Definition and representation of the QoS for multimedia systems. 6th International Conference on High Speed Networking, HPN'95, Palma de Mallorca, Spain, September 11-15, 1995.
16. Chassot C, Fournier M, Lozes A, Diaz M. Service Definition of a Multimedia Partial Order Connection. 2nd COST237 Workshop on Teleservices and Multimedia Communications, Copenhagen, Denmark, November 21-22, 1995.
17. Fournier M, Chassot C, Diaz M, Lozes A. Multimedia Partial Order Transport Architecture: Design and Implementation. Protocol for High Speed Network (PfHSN'96). Sophia Antipolis, November 22-23, 1996.
18. Fournier M, Chassot C, Diaz M, Lozes A. Performance evaluation of a partial order connections. High Perf. Networking (HPN'97). New-York (USA), 28 Avril - 2 Mai 1997.
19. Chassot C, Lozes A, Garcia F, Diaz M, Dairaine L, Rojas L. QoS required by a DIS application in a new generation Internet. IEEE Spring Simulation Interoperability Workshop, Orlando, USA, March 14-19, 1999.
20. Chassot C, Lozes A, Garcia F, Dairaine L, Rojas L. Specification and realization of the QoS required by a distributed interactive simulation in a new generation Internet. LNCS 1718, 6th International workshop on Distributed Multimedia Systems and Telecom. Services (IDMS'99), Toulouse, France, October 12-15, 1999.
21. Garcia F, Chassot C, Lozes A, Diaz M. Conception and evaluation of a QoS based architecture for DIS applications in a differentiated services Internet domain. Optical networking and communications conference (Opticom'01), Denver, USA, August 2001.
22. Garcia F, Chassot C, Lozes A, Diaz M, Anelli P, Lochin E. Conception, implementation and evaluation of a QoS based architecture for an IP environment supporting

differentiated services. 8th International workshop on Interactive Distributed Multimedia Systems (IDMS'01), Lancaster, United Kingdom, September 2001.

23. Chassot C, Garcia F, Auriol G, Lozes A, Lochin E, Anelli P. Performance analysis for an IP differentiated services network. International Communication Conference (ICC'02), New York, USA, May 2002.
24. Chassot C, Auriol G, Lozes A. QoS management protocol for an end to end com. architecture implemented over a differentiated IPv6 network. International Telecommunication Conference (ITC'03), Berlin, Germany, 31 August-5 September 2003.
25. Chassot C, Auriol G, Diaz M. Automatic management of the QoS within an architecture integrating new Transport and IP services in a DiffServ Internet. 6th IFIP/IEEE International. Conference on Management of Multimedia Networks and Services (MMNS'03), Belfast, Ireland, September 7-10, 2003.
26. Auriol G, Chassot C, Diaz M. Toward a signaling architecture in a DiffServ multi-domain environment for per flow end to end QoS. Australian Telecommunications Networks and Applications Conference (ATNAC'04), December 8-10, 2004.
27. Chassot C, Lozes A, Racaru F, Auriol G, Diaz M. A user-based approach for the choice of the IP services in the multi domains DiffServ Internet. 1st International IEEE Workshop on Service Oriented Architectures in Converging Networked Environments (SOACN'06). 18-20 April 2006, Vienna, Austria *[Accepté pour publication le 16/12/2005]*

Conférences francophones avec acte et comité de lecture : 6

28. Chassot C, Diaz M, Lozes A. Principes d'implantation d'une connexion multimédia d'ordre partiel. 4ème Colloque Francophone sur l'Ingénierie des Protocoles (CFIP'95), Rennes, Mai 1995.
29. Fournier M, Chassot C, Diaz M, Lozes A. Evaluation comparative des performances d'une connexion d'ordre partiel et d'une connexion multimédia d'ordre partiel. 6ème CFIP (CFIP'97), Liège, Belgique, 29 Septembre-2 Octobre 1997.
30. Chassot C, Lozes A, Garcia F, Diaz M, Dairaine L. QoS requise par une application de DIS distribuée dans un réseau grande distance. 7èmeCFIP (CFIP'99), Nancy, Avril 1999.
31. Chassot C, Garcia F, Lozes A, Anelli P, Bonald T. Architecture de QoS en environnement IPv6 à services différenciés. 8ème CFIP (CFIP'00), Toulouse, France, Octobre 2000.
32. Garcia F, Auriol G, Chassot C, Lozes A, Lochin E, Anelli P. Conception, implémentation et mesures de performance d'une architecture de com. à QdS garantie dans un domaine IPv6 à services différenciés. 9ème CFIP (CFIP'02), Montréal, Canada, Mai 2002.
33. Auriol G, Chassot C, Diaz M. Evaluation des performances d'une architecture de com. à gestion automatique de la QdS. 10ème CFIP (CFIP'03), Paris, France, Septembre 2003.

Conférences nationales avec acte et comité de lecture : 3

34. Auriol G, Chassot C, Conception d'un protocole de gestion de la Qualité de Service de bout en bout en environnement IPv6 à services différenciés. JDIR 2002, Toulouse, France, Mars 2002.
35. Auriol G, Chassot C. Architecture de communication de bout en bout gérant la QdS dans un environnement DiffServ : étude en simulation ns-2. Colloque de l'École Doctorale Informatique et Télécom. (EDIT'03), Toulouse, France, 14-15 Avril 2003.
36. Auriol G, Chassot C. Architecture de bout en bout à QdS garantie dans un environnement DiffServ multi domaines : protocoles de signalisation. Colloque de l'EDIT (EDIT'04), Toulouse, France, 29-30 Mars 2004.

2. Encadrements

2.1. Résumé

- Thèses soutenues : 2 encadrées à 100%
- Thèse en cours : 2
  - 1 encadrée à 50% (2ème année) – co encadrant Michel Diaz, DR au CNRS
  - 1 encadrée à 75% (1ère année) – co encadrant Khalil Drira, CR (HDR) au CNRS
- Stages de DEA / MASTER Recherche : 5 encadrés à 100%
- Stage de fin d'étude ingénieur : 1 encadré à 100%

2.2. Encadrements de thèse

Fabien Garcia (octobre 1998 à décembre 2002)

- Thèse de doctorat en Informatique de l'EDSYS[9] (INSA de Toulouse)
- Encadrement à 100%. Autorisation d'encadrement accordée par l'INSA de Toulouse
- Sujet : conception, implémentation et mesures des performances d'une architecture de communication de bout en bout à QdS en environnement Internet nouvelle génération
- Mots clefs : architecture, QdS, IntServ, DiffServ, simulation interactive distribuée (DIS)

Guillaume Auriol (octobre 2001 à novembre 2004)

- Thèse de doctorat en Réseau de l'EDIT[10] (INSA de Toulouse)
- Encadrement à 100%. Autorisation d'encadrement accordée par l'INSA de Toulouse
- Sujet : spécification et implémentation d'une architecture de signalisation à gestion automatique de la QdS dans un environnement IP multi domaines
- Mots clefs : architecture, QdS, DiffServ multi domaines, signalisation

Florin Racaru (depuis octobre 2004)

- Thèse de doctorat en Réseau de l'EDIT (INSA de Toulouse)
- Encadrement à 50% avec Michel Diaz, DR au CNRS
- Sujet : signalisation pour la QdS dans l'Internet multi domaines hétérogènes
- Mots clefs : architecture, signalisation, QdS, Internet multi domaines hétérogènes

François Armando (depuis octobre 2005)

- Thèse de doctorat en Réseau de l'EDIT (INSA de Toulouse)
- Encadrement à 75% avec Khalil Drira, CR (HDR) au CNRS
- Sujet : architecture de Transport à politiques auto adaptatives
- Mots clefs : Internet ambiant, architecture dynamique, protocole de Transport dynamiquement configurable, opération d'intervention d'urgence.

[9] EDSYS : Ecole Doctorale Système, Toulouse

[10] EDIT : Ecole Doctorale Informatique et Télécommunication, Toulouse

2.3. Encadrements de stage de DEA/MASTER Recherche (et de fin d'étude ingénieur)

Fabien Garcia (1998), élève ingénieur INSA, spécialité Génie Informatique et Industriel

- DEA Automatique et Informatique Industrielle
- Sujet : conception et implémentation d'un protocole de réservation de ressources pour applications de type DIS en environnement Internet de nouvelle génération (IntServ)

Guillaume Auriol (2001), élève ingénieur ENSICA

- DEA Réseaux et Télécom de l'EDIT
- Sujet : évaluation des performances d'une architecture de gestion de la QdS de bout en bout en environnement IP DiffServ

Florin Racaru (2004), élève ingénieur ENSEEIHT, spécialité Télécom et Réseaux

- DEA Réseaux et Télécom de l'EDIT
- Sujet : spécification UML 2 et implémentation JAVA d'un protocole de signalisation pour la QdS dans un environnement DiffServ multi-domaines

Olivier Roche (2004), élève ingénieur ENSICA

- DEA Réseaux et Télécom de l'EDIT
- Sujet : spécification UML 2 et implémentation JAVA d'une API à QdS dans une architecture DiffServ multi-domaines.

François Armando (2005), élève ingénieur INSA, spécialité Réseaux et Télécom

- MASTER Recherche Informatique, spécialité RTSA[11] de l'EDIT
- Sujet : spécification UML 2 d'un modèle d'architecture de protocole de Transport configurable dynamiquement

Nicolas Rey (1998), élève ingénieur INSA, spécialité Génie Informatique et Industriel

- Projet de fin d'étude
- Sujet : quantification et analyse des dégradations subies par une application de simulation interactive distribuée en environnement IP/ATM

3. Animation scientifique

3.1. Collaborations industrielles

Depuis 1992, nos travaux ont régulièrement été menés dans le cadre de collaborations industrielles, d'abord au niveau national, puis au niveau européen.

Ces collaborations, détaillées ci-après, sont au nombre de cinq :

– CESAME, de 1992 à 1994
– DIS/ATM, de 1997 à 1998
– @IRS, de 1999 à 2001
– @IRS++, de 2001 à 2003
– EuQoS, depuis septembre 2004.

[11] RTSA : Réseaux, Télécom, Système et Architecture

Projet CESAME[12] (1992 à 94 – période de thèse)

- **Nature** : projet national. Financement CNET et CNRS. Durée 24 mois.
- **Partenaires** : CNET, CCETT et neuf laboratoires du CNRS dont le LAAS-CNRS
- **Objectif** : le projet avait pour but de développer des architectures, des méthodes et des techniques, permettant de concevoir des applications multimédia et coopératives. Il comportait cinq thèmes dédiés : (1) au développement d'applications du type visioconférence ; (2) à l'étude des échanges dans le cadre de groupes coopératifs d'utilisateurs ; (3) à l'analyse des contraintes de synchronisation, en particulier temporelles, dans les objets et les systèmes de communication multimédia ; (4) à l'étude et à la proposition de nouveaux services et protocoles de Transport, utilisant des réseaux à hauts débits ; (5) à la représentation formelle des mécanismes et des protocoles, pour l'évaluation de leurs performances et de leur test temporel.
- **Responsabilité dans le projet** : aucune (période de thèse).
- **Contribution au projet** : nos travaux ont adressé le thème des nouveaux services et protocoles de Transport ; ils ont conduit à la proposition et à la spécification formelle (dans le langage ESTELLE) du concept de *connexion de Transport à ordre et fiabilité partiels*, présenté dans l'Etape 1 de ce mémoire.

Projet DIS/ATM (1996 à 1998)

- **Nature** : projet national. Financement DGA[13] et MESR[14]. Durée 24 mois.
- **Partenaires** : Dassault Electronique, INRIA Sophia Antipolis, LIP6, LAAS-CNRS
- **Objectif** : le projet avait pour but d'identifier les problèmes posés par l'emploi des protocoles et des mécanismes de l'Internet nouvelle génération (IPv6 et IntServ/RSVP) sur une infrastructure ATM locale et nationale, et d'étudier les solutions permettant d'offrir une QdS de bout en bout compatible avec les besoins des applications de simulation interactive distribuée (DIS)
- **Responsabilité dans le projet** : responsable scientifique pour le LAAS avec M. Diaz de la parte relative à l'interfaçage de DIS avec le système de communication.
- **Contribution au projet** : nos travaux ont adressé la conception et l'implémentation de l'architecture de bout en bout du projet. Ces travaux ont conduit aux résultats suivants : (1) la caractérisation de la QdS requise par une application DIS déployée dans un environnement réseau grande distance ; (2) la conception et l'implémentation d'une architecture : offrant une QdS en accord avec les besoins des applications DIS, et abstrayant à ces applications la complexité d'utilisation du protocole RSVP, nécessaire à la mise en œuvre des services IntServ. Cette architecture a été déployée et évaluée par des mesures de performance sur la plateforme nationale issue du projet.
- **Encadrement** : ces travaux ont fait l'objet de la première partie de la thèse de F. Garcia.

---

[12] CESAME : Conception formElle de Systèmes hAuts débits Multimédia coopEratifs

[13] DGA : Direction Générale de l'Armement

[14] MNESR : Ministère Enseignement Supérieur et de la Recherche

Projet @IRS[15] (1999 à 2001)

- **Nature** : projet national du RNRT[16] - Financement CNRS et MESR. Durée 27 mois.
- **Partenaires** : Thomson CSF-Detexis, ELV-EADS, CNET, LAAS-CNRS, INRIA Sophia, LIP6, LSIIT, IMAG-LSR
- **Objectif** : le projet avait pour but de définir une Architecture Intégrée de Réseaux et de Services Internet de nouvelle génération répondant aux derniers standards de l'Internet (IPv6), et s'attaquant au problème de la QdS et de la mobilité. Initié en décembre 1998, le projet s'est terminé avec succès en février 2001, il est à l'origine de la mise en place de la plate-forme @IRSbone. Nos travaux ont adressé la partie QdS.
- **Responsabilité dans le projet** : responsable scientifique pour le LAAS avec M. Diaz des sous projets 2 (Qualité de service) et 5 (Plateforme de simulation interactive).
- **Contribution au projet** : nos travaux ont adressé la conception, l'implémentation et la mesure des performances d'une architecture offrant une QdS de bout en bout par flux dans un environnement IP DiffServ mono domaine (offrant donc en cœur une QdS par agrégat). Cette architecture a été déployée dans le cadre d'une plateforme nationale (le @IRSBone) basée sur une infrastructure ATM.
- **Encadrement** : ces travaux ont fait l'objet de la deuxième partie de la thèse de F. Garcia.

Projet @IRS++ (2001 à 2003)

- **Nature** : projet national du RNRT - Financement du MESR. Durée 27 mois.
- **Partenaires** : 6 WIND, RENATER, EADS, REALIX, LAAS, EURECOM, INRIA Sophia, LIP6, LSIIT, IMAG LSR
- **Objectif** : suite directe de @IRS, le projet @IRS++ abordait de nouveaux problèmes : (1) l'intégration des réseaux fixe/mobile, afin de disposer d'un accès à ses applications ou ses informations quel que soit le lieu et le mode d'accès ; (2) les communications de groupe, pour échanger, partager et travailler au sein d'une communauté d'utilisateurs ; (3) la mise en place dynamique de nouveaux services à la demande des usagers ou suite à de nouvelles offres d'opérateurs.
- **Responsabilité dans le projet** : contribution à l'élaboration du projet et responsable scientifique pour le LAAS, avec M. Diaz, du sous projet 4 (Programmabilité des services).
- **Contribution au projet** : nos travaux ont adressé le besoin en services programmables à QdS garantie (problème 3 du projet). Ils ont conduit à l'extension du l'architecture @IRS par : (1) l'intégration d'un Transport à fiabilité partielle[17] et (2) la conception, l'implémentation et l'intégration d'un algorithme de sélection du meilleur couple (Transport, IP) permettant de satisfaire une requête de QdS exprimée en des termes génériques, sans que l'application ait à expliciter les services à activer.
- **Encadrement** : ces travaux ont fait l'objet de la première partie de la thèse de G. Auriol.

---

[15] @IRS : Architecture Intégrée de Réseaux et Services

[16] RNRT : Réseau National de la Recherche en Télécommunications

[17] FPTP : *Fully Programmable Transport Protocol*, conçu et développé au LAAS et à l'ENSICA [Expo03].

Projet EuQoS[18] **(2004-)**

- **Nature** : projet IST du 6ème PCRD (*Integrated project*). Financement de la communauté européenne. Durée : 36 mois en deux phases de 18 mois chacune
- **Objectif** : le projet (en cours) a pour but de proposer, développer, intégrer, tester et valider toutes les technologies liées à l'obtention de QdS de bout en bout sur une infrastructure de réseaux multiples et hétérogènes déployés à l'échelle européenne. Le système devra supporter plusieurs applications à QdS (voix sur IP, visioconférence, télé ingénierie, etc.), conscientes ou non des capacités en QdS du réseau.
- **Partenaires** : 25 partenaires européens, industriels et académiques, parmi lesquels plusieurs opérateurs réseaux, notamment Telefonica ID (leader) et France Télécom.
- **Responsabilité dans le projet** : contribution à l'élaboration du projet. Responsable scientifique LAAS, avec M. Diaz, de la partie signalisation des *work packages* 1 et 3 respectivement dédiés à la conception et à l'implémentation de l'architecture EuQoS. Correspondant pour le LAAS du *work package* 6 concernant la dissémination des résultats et à la mise en place des cours qui seront délivrés dans le cadre du projet.
- **Contribution au projet** : nos travaux adressent le besoin en signalisation inter domaines pour la QdS. Notre contribution porte sur (1) la conception, la spécification et l'implémentation d'un protocole de signalisation découplé du chemin de donnée, et (2) son intégration dans l'architecture globale définie pour la 1ère phase du projet ; pour la 2ème phase, nos travaux ont pour objectif de tenir compte de la nature hétérogène des domaines de l'Internet vis à vis de la gestion de la QdS multi domaines.
- **Encadrement** : ces travaux font l'objet de la thèse (en cours) de F. Racaru.

3.2. Thème SCD de la fédération inter laboratoires FERIA

Depuis 2000, je suis co-responsable pour le LAAS du thème Simulation Coopérative Distribuée (SCD[19]) de la fédération inter laboratoires FERIA (CERT-ONERA, IRIT, LAAS).

Objectif : aborder sous forme de séminaires, de réunions de travail et d'expérimentations entre les sites, trois types de problèmes : la simulation, les systèmes d'exploitation distribués, les plates-formes à haut débit et la mise en œuvre sur des réseaux ouverts de type Internet.

Contribution au thème : participation aux réunions du groupe (une par mois), séminaires dispensés, participation aux expérimentations, rédaction des rapports d'activités du thème.

3.3. Séminaires invités

- Séminaire à l'Université de Delaware (Etats Unis), 2002
  - Thème : *Performance Analysis for an IP Differentiated Services Network*
- Séminaire à l'école d'été temps réel ETR'03, IRIT, Toulouse, 2003
  - Thème : Simulation Interactive distribuée et Internet à QdS
- Séminaire du Pôle Systèmes Informatiques Critiques (SYNC) du LAAS-CNRS, 2003
  - Thème : Nouveaux réseaux et protocoles

[18] EuQoS : End-to-end Quality of Service support over heterogeneous networks

[19] Le responsable général du thème est P. Siron (CERT-ONERA)

3.4. Participation à jury de thèse

- Soutenance de thèse de B. Breholee (CERT-ONERA, Toulouse, mars 2005)
  - Sujet : Simulation distribuées et domaines
  - Directeur de thèse : P. Siron, CERT-ONERA
- Soutenance de thèse de N. Larrieu (LAAS-CNRS, Toulouse, juillet 2005)
  - Sujet : Contrôle de congestion et gestion du trafic à partir de mesures pour l'optimisation de la QdS dans l'Internet
  - Directeur de thèse : P. Owezarski, CR au LAAS-CNRS

III. Activités d'enseignement

1. Introduction

J'ai été recruté en tant que Maître de Conférences au Département Génie Electrique et Informatique (DGEI) de l'INSA de Toulouse en septembre 1996, sur un profil Informatique et Réseaux. Les 5 années de formation des ingénieurs INSA étaient alors réparties en 2 années de premier cycle (équivalentes aux classes préparatoires) et 3 années de spécialisation dans les différents départements, dont le DGEI.

Note : ce schéma a été modifié en 2003 suite à la réforme du cursus général INSA en : 1 année de tronc commun puis 2 années de pré orientations et enfin 2 années de spécialisation. Afin simplifier les explications qui suivent, je présente mes implications dans l'ancien schéma.

En 1996, le DGEI comportait 2 spécialités : « Génie Informatique et Industriel » (GII) et « Automatique, Electronique Informatique » (AEI), comportant chacune 2 orientations de 5ème année (Figure 1).

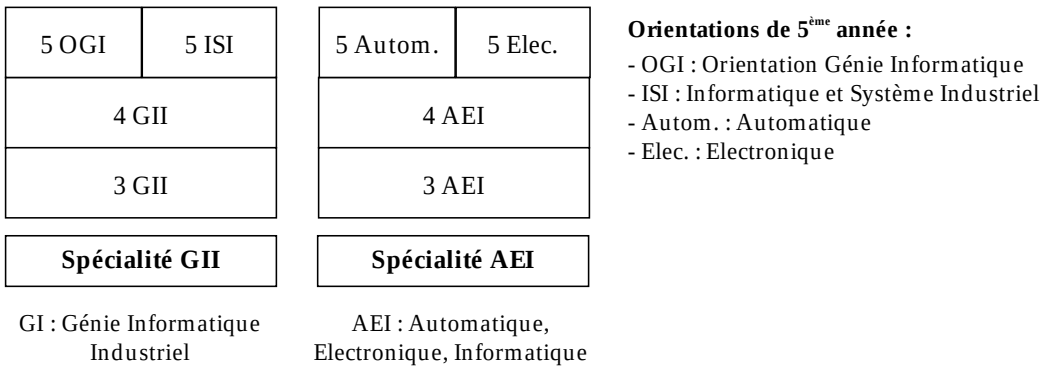


Figure 1 : Schéma des spécialités du DGEI à mon arrivée en 1996/97

En tant que premier enseignant recruté à l'INSA sur un profil Réseaux et Informatique, il m'a été confié, dès mon arrivée, plusieurs responsabilités pédagogiques visant à faire évoluer les enseignements relatifs aux réseaux informatiques, d'abord à l'échelle du département GEI, puis à celle de l'INSA.

J'ai également été chargé d'un certain nombre d'autres responsabilités : de 5ème année, de MASTER et de Conseil d'Orientation.

Ces différentes responsabilités sont décomposables en trois volets majeurs :

1) Prise en charge et évolutions des cours réseaux au DGEI

A mon arrivée, les cours réseaux du DGEI étaient au nombre de deux (« Introduction aux réseaux » et « Réseaux locaux industriels »), effectués par des vacataires extérieurs.

J'ai eu à reconstruire et à étendre ces deux cours[20], et à en créer quatre autres, chronologiquement : « Technique de description formelle de protocoles : Estelle », « Programmation réseaux : socket TCP/UDP », « Interconnexion de réseaux et routage » et « Architecture de communication pour le multimédia et qualité de service dans l'Internet ».

Ces différents cours sont décrits ci-après en section 2.

2) Elaboration de nouveaux programmes d'enseignement au DGEI et à l'INSA

Une implication forte dans mon investissement à l'INSA concerne l'élaboration de nouveaux programmes d'enseignement, d'abord dans le cadre d'une **nouvelle orientation de 5ème année au DGEI**, puis dans celui d'une **nouvelle spécialité de 3 ans à l'INSA**, dédiées chacune aux réseaux et télécommunications :

- en 1998, le DGEI a décidé de créer 2 nouvelles orientations de 5e année, en remplacement des orientations ISI et Automatique. La responsabilité qui m'a été confiée a été d'élaborer et de coordonner la mise en œuvre de la partie Informatique et Réseaux du programme d'enseignement de l'orientation « Réseaux et Télécommunications », accessible par les étudiants des deux spécialités (AEI et GII) ; la première promotion est sortie en juin 1999.
- en 2001, et pour faire face à la forte demande industrielle, il a été décidé au niveau de l'INSA de créer une nouvelle spécialité de 3 ans à l'INSA, entièrement dédiée aux « Réseaux et Télécommunications ». La responsabilité qui m'a été confiée a là encore été d'élaborer et de coordonner la mise en œuvre de la partie Informatique et Réseaux du programme des 3 années de la spécialité ; la première promotion est sortie en juin 2004.

Les programmes que j'ai coordonnés sont présentés plus loin en section 3.

3) Autres responsabilités

Parallèlement, j'ai eu plusieurs autres responsabilités, qui sont les suivantes :

- depuis mon arrivée, j'ai assumé **6 années de responsabilité de 5ème année** :
  - 4 ans en 5ème année Génie Informatique de 1996/97 à 1999/00
  - 2 ans en 5ème année Réseaux et Télécommunications (RT) en 2003/04 et 2004/05

  Cette responsabilité impose d'établir l'emploi du temps sur le semestre de présence des étudiants (d'octobre à février[21]) et d'assurer la présidence de tous les jurys de soutenance de projet de fin d'étude (en juin) ;
- depuis 2002/03, j'assume la **responsabilité INSA de la spécialité RTSA du MASTER Recherche Informatique** (ex DEA Réseaux et Télécommunications) ;

[20] Le second est désormais effectué par un autre enseignant, plus proche des réseaux locaux de part ses activités de recherche.

[21] Les étudiants partent ensuite en stage de fin d'étude.

- depuis 2004, j'ai la **responsabilité du Conseil d'Orientation de la spécialité RT**, dont le rôle est d'assurer la coordination et l'évolution du programme des 3 années de la spécialité ;
- depuis 1998, je suis **membre élu de la commission de spécialiste 61-63** ;
- enfin, de 1997 à 2003, j'ai été **membre élu du conseil de département GEI**, dont le rôle principal et de statuer sur les décisions locales au département.

Je présente maintenant plus en détails les enseignements que je dispense ainsi que ma contribution à l'élaboration de nouveaux programmes d'enseignements.

2. Enseignements

2.1. Résumé

Mon activité d'enseignement concerne essentiellement les **réseaux informatiques**, **incluant de la programmation**. J'enseigne de façon régulière :

- à l'**INSA de Toulouse** de puis 1994 (vacataire puis ATER puis Maître de Conférences)
  - en 3ème année[22] RT, GI[23] (ex GII) et AEI ;
  - en 4ème et 5ème année en RT, GI et AEI[24].
- dans la **spécialité RTSA[25] du MASTER Recherche Informatique**[26] de l'EDIT (depuis 2000)

Les principaux enseignements que je dispense (ou que j'ai dispensés) sont résumés ci-après (Tableau 1). Les enseignements précédés d'un **(*)** sont ceux que j'ai créés.[27].

Intitulé du cours	Type (CM/TD/TP) et volume (étudiant)	Année d'étude	Années de dispense
(*) Introduction aux réseaux	15H CM + 2,5H TD + 6 TP	3RT, 3GI, 3AEI	depuis 1996
(*) Interconnexion de réseaux locaux routage et Transport	20H CM	4RT, 4AEI, 4GI MASTER RTSA	depuis 1998
(*) Architecture multimédia et QdS	10H CM	5RT MASTER RTSA	depuis 2003
(*) Programmation réseau (socket)	5H CM + 6H TP	3RT, 3 GI, 3AEI	depuis 1996
(*) Projets tuteurés	60H TP (20h encadrant)	4RT	depuis 2002
(*) Techniques de description formelle (ESTELLE)	10H CM + 12H TP	5GI	1994 à 1998
Algorithmique et Prog. (ADA)	48H TP	3GI	1996 à 2002

Tableau 1 : Résumé des cours créés et/ou dispensés

[22] Au travers, dans le cadre de la réforme de 2003, des pré-orientations correspondantes.

[23] La spécialité « Génie Informatique Industriel » est devenue « Génie Informatique » à partir de 2002.

[24] RT : Réseaux et Télécom / GI : Génie Informatique / AEI : Automatique et Electronique Intégrée

[25] RTSA : Réseau, Télécom, Système et Architecture

[26] Auparavant DEA Réseaux et Télécommunication (de 2000 à 2003)

[27] Je participe également à des modules d'ouverture de quelques heures chacun sur les réseaux et l'Internet.

2.2. Détails des enseignements

Les détails des enseignements introduits dans le tableau précédent sont les suivants :

- **Introduction aux réseaux informatiques (depuis 1996/97)**
  - **Type** : CM (15h étudiant), TP (6h étudiant), TD (2h30 étudiant)
  - **Niveau** : 3e année (RT, GI, AEI)
  - **Historique** : j'ai créé cet enseignement à mon arrivée à l'INSA en 1996/97 (le cours était effectué auparavant par des vacataires extérieurs qui n'ont laissé ni support ni sujet/réalisation de travaux pratiques). Un support de cours de 180 pages est fourni aux étudiants. C'est le cours de base de la formation en réseau de l'INSA.
  - **Description** : l'objectif est d'acquérir les connaissances fondamentales liées à la conception des réseaux informatiques, avec une focalisation sur leur modèle d'architecture. La première partie du cours introduit les caractéristiques des principales applications d'un réseau (http, ftp, …). La deuxième partie détaille les principaux concepts des réseaux : connectivité, partage des ressources, commutation, qualité de service et principalement architecture (services, protocoles, modèle en couches avec comme illustrations principales le modèle OSI et l'architecture de l'Internet TCP/IP). La troisième partie du cours détaille l'architecture des réseaux locaux (LLC/MAC) avec étude de cas aux réseaux Ethernet et application en TD. Deux séances de TP d'analyse de trafic TCP/IP illustrent les concepts introduits en cours.
- **Interconnexion de réseaux locaux, routage, Transport dans l'Internet (depuis 1998/99)**
  - **Type** : CM (20h étudiant en RT, 15h étudiant en GI et AEI)
  - **Niveau** : 4e année INSA (RT, GI, AEI) et MASTER Recherche RTSA (10h sur les 15)
  - **Historique** : j'ai créé ce cours à l'occasion de la création de l'orientation de 5e année Réseaux et Télécommunications en 1998/99 (ce cours n'existait pas auparavant). Un support de cours de 210 pages est fourni aux étudiants *[Ce support de cours est joint au dossier à titre illustratif de la forme des polycopiés que je distribue aux étudiants]*.
  - **Description** : le cours présente d'abord les solutions classiques (répéteur, pont, routeur) permettant d'interconnecter des réseaux locaux. Il détaille ensuite les algorithmes de routage mis en œuvre d'abord dans une interconnexion par ponts (*spanning tree* et *source routing*), puis par routeurs IP (routage IP, *subnetting* et routage sans classe) ; il introduit également les protocoles de routage attenants (RIP, OSPF, BGP). Le cours présente ensuite les principales évolutions des solutions d'interconnexion de niveaux 2 et 3 : réseaux locaux virtuels (VLAN Ethernet), proxy applicatif, translation d'adresse réseau (NAT), réseaux privés virtuels (VPN), et introduit enfin les évolutions de l'Internet (IPv6n QdS, …), avec IP multicast comme cas d'étude. Enfin, le cours détaille les deux principaux protocoles de Transport de l'Internet (TCP et UDP).
  - **Note** : neuf séances de travaux pratiques (27h au total) orientées administration de réseaux (Ethernet et IP) sur PC Linux et Windows illustrent le cours ; ces TP sont effectués par un autre enseignant du DGEI (spécialiste de l'administration réseau) avec lequel j'ai défini le sujet des TP lors de la création de l'orientation de 5ème année.

- **Architecture de communication multimédia et QdS dans l'Internet (depuis 2003/04)**
  - **Type** : CM (10h étudiant)
  - **Niveau** : 5e année INSA (RT) et MASTER Recherche RTSA
  - **Historique** : j'ai créé cet enseignement à l'occasion de la création de la 5ème année de la spécialité Réseaux et Télécommunication en 2003/04. Un support de cours de 130 pages est délivré aux étudiants.
  - **Description** : Face aux besoins en qualité de service (QdS) des nouvelles applications distribuées dans l'Internet, plusieurs solutions ont été proposées pour repousser les limites des protocoles de l'Internet (niveaux Transport et IP). Le cours explore les solutions actuelles et introduit les solutions futures qui font l'objet des travaux de recherche les plus récents. Il est structuré en deux parties principales : la première illustre les solutions envisagées pour permettre aux applications de s'adapter aux variations des performances de l'Internet actuel sans QdS (RTP/RTCP, mécanismes d'absorption de gigue et de récupération des pertes par FEC et entrelacement) ; la deuxième partie présente les propositions de recherche visant à ce que l'Internet offre des garanties de QdS aux applications (protocoles de Transport à ordre et fiabilité partiels, IntServ, DiffServ mono/multi domaines, MPLS).
- **Programmation réseau : socket TCP/TCP en C (depuis 1996/97)**
  - **Type** : CM (5h étudiant), TP (6h étudiant)
  - **Niveau** : 3e année INSA (RT, GI, AEI)
  - **Historique** : j'ai créé cet enseignement à mon arrivée à l'INSA en 1996/97 (ce cours n'existait pas auparavant). Un support de cours de 70 pages est fourni aux étudiants.
  - **Description** : le cours décrit l'interface de programmation (API) la plus utilisée pour programmer la partie communication d'une application distribuée : l'API socket TCP/UDP (ici en C). Le cours décrit également les principaux mécanismes sous jacents à la fiabilisation du transfert de données entre deux hôtes. Deux séances de TP illustrent le cours par la programmation d'une application distribuée de type client/serveur (en l'occurrence, un générateur de trafic). Note : ces TP sont prolongés par un bureau d'étude de 12h (TP) durant lesquels les étudiants implémentent en C les différents types de protocoles de reprise des pertes présentés dans le cours.
- **Projet tuteuré (depuis 2002/03)**
  - **Niveau** : 4e année INSA (RT)
  - **Type** : les projets tuteurés ont été introduits au DGEI en 2002/03. Il s'agit de projets longs (pour des groupes de 4 à 5 étudiants) s'étendant sur l'année à raison de 60h étudiants (comptées 20h équivalent TP pour l'encadrant).
  - **Description** : ces trois dernières années, j'ai co encadré 3 projets intégrant successivement les résultats du projet de l'année précédente ; les sujets proposés sont en relation directe avec le cours sur la QdS que j'effectue en 5ème année RT et avec mes activités de recherche ; deux des étudiants impliqués dans les projets ont ensuite poursuivi en recherche sous mon encadrement (un qui démarre sa thèse cette année, et un autre dont le stage de DEA débutera en février 2006).
    - ***Sujet du projet 1*** : Développement en JAVA d'une application de streaming vidéo et mise en œuvre sur une plate forme d'émulation ;

- ***Sujet du projet 2*** : Développement en JAVA d'une application de visioconférence et mise en œuvre sur une plate forme RSVP ;
- ***Sujet du projet 3*** : Application multimédia et QdS dans l'Internet IntServ/ DiffSer**v.**

- **Techniques de description formelle de protocoles : ESTELLE (en 1994/95 à 1997/98)**
  - **Type** : CM (10h étudiant), TP (12h étudiant)
  - **Niveau** : 5e année GII
  - **Historique** : Cet enseignement est le premier que j'ai créé à l'INSA, durant la période de thèse en tant que vacataire à l'INSA (1994/95). Un support de cours de 100 pages était délivré aux étudiants. Le langage Estelle ne s'imposant pas dans le monde industriel, j'ai mis fin à la dispense de ce cours à l'occasion de la création de l'orientation Réseaux et Télécommunication en 1998/99.
  - **Description** : Estelle est une technique de description formelle de services et de protocoles de communication. Les entités de protocole y sont modélisées par des machines à états étendues, communicant de façon asynchrone par file FIFO. Le cours présente le contexte d'utilisation d'Estelle, ses principaux concepts, sa syntaxe et les différents comportements d'une spécification Estelle. Quatre séances de TP illustrent le cours par la spécification d'un sous ensemble du protocole de niveau MAC IEEE 802.4.
- **Algorithmique et programmation (de 1996/97 à 2001/ 02)**
  - **Type :** TP (48h étudiant)
  - **Niveau** : 3e année GII
  - **Description** : les TP se déroulent à raison d'une séance de trois heures par semaine durant toute l'année. De façon non exhaustive, les sujets de TP portent principalement sur les sous-programmes, les mécanismes d'exception propres à ADA, les paquetages, la récursivité, les piles, les pointeurs et la manipulation de fichiers.
  - **Historique** : cet enseignement existait à mon arrivée. Je n'ai eu qu'à le dispenser.

3. Elaboration de nouveaux programmes d'enseignement

3.1. Orientation de 5e année Réseaux et Télécommunications

A mon arrivée, le DGEI comportait deux spécialités (GII et AEI) donnant chacune accès à deux orientations de 5ème année. Ce schéma a été revu en 1997, conduisant au remplacement de deux des orientations existantes par deux nouvelles orientations (Figure 2) :

« **Réseaux et Télécommunications (RT)** » et « Temps Réel et Système (TRS) »

J'ai eu la charge de la partie informatique et réseaux du programme de la 5ème année RT.

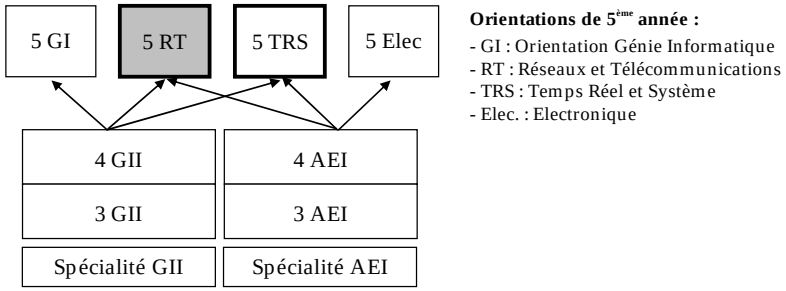


Figure 2 : Schéma des orientations de 5ème année au DGEI (en 1998/99)

Le programme a été élaboré autour de trois thèmes de 100 heures étudiants chacun :

- **Systèmes distribués multimédia et coopératifs**, traitant des moyens de réalisation de logiciels distribués de niveau applicatif et middleware ;
- **Réseaux et ingénierie des protocoles**, portant sur les réseaux proprement dits (réseaux locaux, réseaux à grande distance, réseaux à hauts débits, interconnexion IP,…) ainsi que sur la modélisation et l'évaluation des performances des protocoles ;
- **Télécommunications**, concernant les aspects relatifs à l'électronique, au traitement du signal, et aux architectures de circuits et de dimensionnement des liaisons.

Ma contribution a porté sur l'élaboration et sur la mise en œuvre des deux premiers thèmes.

3.2. Spécialité Réseaux et Télécommunications

Face à la demande croissante en ingénieurs RT, il a été décidé en 2001 de créer une nouvelle spécialité à l'INSA de Toulouse, portant intégralement sur ce thème (Figure 3). Suite à la création de l'orientation de 5ème RT, c'est au DGEI qu'est revenue la responsabilité d'élaborer le programme d'enseignement de cette spécialité.

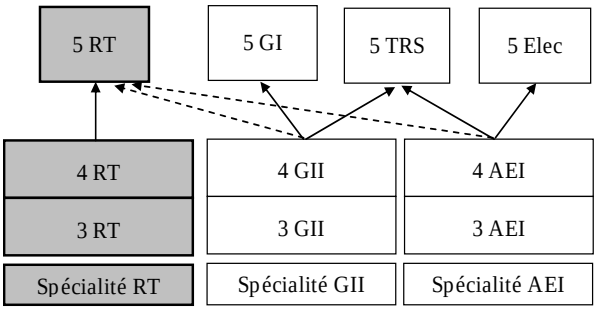


Figure 3 : Schéma des spécialités du DGEI (en 2002/03)

En collaboration avec B. Pradin (Directeur du DGEI à cette époque) et J.M. Dilhac (Professeur 63e chargé des aspects Electronique et Télécommunications, et aujourd'hui Directeur du DGEI), **j'ai été le porteur du projet pour les aspects Informatique et Réseaux**.

Le programme d'enseignement de la spécialité RT comporte 540h scientifiques[28] par année équilibrées entre l'informatique, les réseaux, l'électronique et les télécommunications :

- **la 3ème année**, qui pose les bases en informatique et en électronique, a été en grande partie conçue à partir des programmes d'enseignement existants en 3GII et 3AEI ;
- **la 4ème année**, plus orientée informatique, réseaux et télécommunications, est bâtie sur la base du programme de la 5ème RT définie avant la création de la spécialité ;
- **c'est sur la 5ème année, entièrement nouvelle, que s'est portée la partie la plus importante de mon travail ; le programme est conçu autour de quatre modules de 65 à 90h chacun :**
  - le premier module approfondit les cours de 4ème année pour couvrir les compétences de base requises pour un ingénieur RT ;
  - les trois autres modules sont des **modules thématiques (MT)**, dont l'objectif est de traiter un thème spécifique des réseaux et télécommunications :

[28] C'est à dire, hors langues vivantes, gestion de projet, *marketing*, …

1) le MT **Multimédia** traite de la mise en œuvre des applications multimédia dans l'Internet : standards de composition de documents, interfaces de programmation audio/vidéo, solutions aux besoins en QdS et en communication de groupe ;
2) le MT **Circuits pour les télécommunications mobiles**, traite des technologies logicielles et matérielles mises en œuvre dans les systèmes de téléphonie mobile de 3ème génération, pour le transport des données multimédia.
3) le MT **Ingénierie des réseaux** traite de l'évaluation de performances appliquées aux réseaux, de la planification et de l'optimisation de réseaux d'opérateurs, ainsi que de la caractérisation et de l'analyse du trafic dans l'Internet (métrologie).

Ces modules s'appuient sur les compétences des groupes de recherche concernés du LAAS et du LESIA[29]. Le programme de la spécialité RT est disponible sur le site web de l'INSA[30].

* * *

[29] Le LESIA (Laboratoire d'Etude des Systèmes Informatiques et Automatiques) est un laboratoire propre de l'INSA situé au DGEI (http://www.lesia.insa-toulouse.fr/index.php).

[30] http://www.insa-toulouse.fr/formation/ingenieur/specialites/RT/RT_fr.htm

Partie 2

Travaux de Recherche

INTRODUCTION

Contexte

Depuis quinze ans, les évolutions de l'informatique et des télécommunications ont conduit à une modification radicale du paysage de la communication informatique, et en conséquence de l'Internet et de ses services.

Initialement basées sur l'échange de données textuelles (transfert de fichiers, mail, …), les applications distribuées manipulent à présent tous les types de médias (en premier lieu l'audio et la vidéo), et sont potentiellement multi utilisateurs, coopératives et dynamiques[31] : les systèmes de visioconférence, de partage d'applications ou d'enseignement à distance sont autant d'exemples d'applications maintenant courantes dans l'Internet ;

Parallèlement, les premiers réseaux, à faibles performances[32], ont fait place à des technologies nouvelles, d'abord filaires, offrant à échelle locale ou en grande distance, de hauts débits et des taux d'erreurs quasi nuls, et ensuite sans fil, en grande distance (satellite) puis en local, autorisant la mobilité des utilisateurs autour des points d'accès (WiFi, …).

Forts de cette dernière diversité, les services de communication de l'Internet se sont ainsi étendus : ils sont désormais accessibles depuis plusieurs types de terminaux (PC, PDA, …) et de points d'accès (filaire ou sans fil), et les débits offerts par les fournisseurs sont dorénavant compatibles avec le téléchargement de gros volumes de données (vidéo, musique, …).

Problématique générale

En dépit de ces avancées, ces services sont encore très insuffisants : les besoins des applications ne portent pas uniquement sur le débit, mais aussi et surtout sur le délai d'acheminement de leurs échanges, par connexion ou par paquet.

Sont ainsi concernées toutes les applications interactives impliquant deux ou plusieurs utilisateurs collaborant à une tache commune : pour ces applications, du type téléphonie sur Internet ou jeux distribués en réseau notamment, il est indispensable que les utilisateurs aient une perception *intelligible* du contexte, et la maîtrise du délai des transmissions est incontournable. La cohérence requise par ces applications nécessite des services de communication à la fois plus performants pour différents critères, mais également beaucoup plus dynamiques, c'est à dire pour le moins programmables, et idéalement auto-adaptatifs.

Les services offerts à l'heure actuelle sont très loin de cet objectif et constituent un frein au déploiement des applications futures, qui impliqueront une dynamicité encore plus forte des collaborations entre utilisateurs.

C'est dans cette problématique générale de la garantie de ***Qualité de Service (QdS)*** *de bout en bout dans l'Internet, que s'inscrivent nos travaux de recherche depuis 1992. Ces travaux adressent le besoin de nouveaux services, protocoles et architectures de communication pour l'Internet, dans le but d'offrir des garanties de QdS aux nouvelles applications distribuées.*

[31] Par exemple, pour l'entrée ou la sortie de participants durant le déroulement de l'application.

[32] En termes de débit d'accès, de taux d'erreurs bits, …

La difficulté de cette problématique est double. Elle résulte :

- d'une part, de la différence et de la variabilité des besoins applicatifs,
- d'autre part, de la structure de l'Internet, aujourd'hui hétérogène à plusieurs niveaux.

La première difficulté s'exprime en termes de modélisation des besoins applicatifs et de leur évolution, indispensable pour appréhender la conception de nouveaux protocoles de communication, ou l'utilisation de services génériques programmables.

L'Internet est par définition hétérogène : pendant un temps, cette hétérogénéité s'est exprimée au seul niveau des technologies réseau interconnectées, offrant des services et des performances différentes. Ces dernières années, l'explosion de l'Internet a vu le développement de nouveaux services, tant de niveau applicatif que de communication, fournis par de nouvelles entités : les *opérateurs*[33] de l'Internet.

Ainsi, la structure de l'Internet, est aujourd'hui non seulement *multi réseaux*, mais également *multi domaines,* assurant chacun et de façon indépendante, la gestion de leurs ressources. La difficulté qui en résulte est liée à la nécessité d'aborder le problème de la QdS, non plus seulement à la seule échelle du multi réseaux, mais également à celle du multi domaines, assurant des services hétérogènes.

Thématiques de recherche

Dans ce contexte et face à cette problématique, nos contributions ont adressé, en visant à intégrer les solutions successivement proposées, trois thèmes de recherche consécutifs :

- **les nouveaux services et protocoles de Transport pour optimiser la QdS** en satisfaisant les contraintes minimales de l'application ; ce thème a été adressé durant la période de thèse et l'année après thèse ;
- **les architectures de bout en bout pour garantir la QdS,** en intégrant les nouveaux services de l'Internet, aux niveaux Transport et IP (IntServ puis DiffServ) ;
- **la signalisation pour la QdS**, nécessaire pour aborder l'hétérogénéité de l'Internet à l'échelle du multi domaines, dans l'optique d'une garantie de QdS de bout en bout.

Notre approche a toujours été de partir de l'expression des besoins applicatifs, et de définir ensuite comment optimiser les performances ou garantir la QdS, en satisfaisant les contraintes minimales exprimées par l'application.

L'évolution de nos thématiques est liée à celle des réseaux et de l'Internet.

Nos contributions aux services et protocoles de Transport (1992 à 1996) ont été menées à une époque où l'Internet, dit *best effort,* n'offrait comme QdS que celle fournie de bout en bout par TCP[34]. La garantie au niveau IP n'était pas encore considérée, car se dessinait la promesse d'un déploiement massif de la technologie ATM.

La problématique que nous avons adressée concernait les limites conceptuelles des protocoles existants, au regard des caractéristiques des nouvelles applications multimédia.

[33] Fournisseurs d'accès, de service, ou de transit.

[34] TCP garantit la délivrance des données dans leur ordre d'émission, sans tenir compte des aspects temporels (délai, gigue).

Nos contributions aux architectures de bout en bout à QdS font suite aux propositions des groupes IntServ[35] puis DiffServ[36] de l'IETF, visant à définir comment offrir de nouveaux modèles de services pour le niveau IP [Brad97a] [Brad97b] [Shen97] [Wroc97] [Blak98].

La problématique que nous avons adressée était de définir comment coupler de façon cohérente les nouveaux services de niveau Transport[37] et de niveau IP de l'Internet, en minimisant l'utilisation des ressources, et en visant à rendre transparent le système de communication aux programmeurs d'applications.

Nous avons successivement considéré les contextes :

- IntServ (en 1997 et 1998) pour une famille d'applications complexes, la simulation interactive distribuée (DIS[38]), incluant les jeux distribués ;
- DiffServ, dans un contexte d'abord mono domaine (de 1999 à 2001), puis multi domaines (à partir de 2002), pour des applications génériques à large spectre de besoins en QdS.

Nos contributions à la signalisation (à partir de 2003) s'inscrivent dans le cadre de nos travaux d'architecture pour la QdS dans l'Internet multi domaines. Cette signalisation est nécessaire à plusieurs fonctions comme le test de la disponibilité de bande passante sur le chemin emprunté par les données[39].

La problématique que nous avons adressée part de notre choix de modèle d'Internet à QdS, à base d'entités logiques (BB[40]) gérant les ressources de chaque domaine, et qui a pour conséquence un découplage entre le chemin emprunté par les données de signalisation et celui des données applicatives. Le problème qui se pose alors est de définir comment, d'une part, assurer la communication entre les BB qui ne se connaissent pas a priori, et d'autre part, conduire à des réservations concernant des équipements impliqués dans le chemin[41] des données applicatives[42] qui ne sont pas conscients des échanges de signalisation entre BB.

Plan du mémoire

La suite de cette deuxième partie du mémoire est structurée comme suit :

- un état de l'art des trois thématiques de recherche abordées est tout d'abord fourni ;
- nos travaux sont ensuite présentés en trois étapes :
  - l'étape 1 résume nos contributions aux services et protocoles de Transport ;
  - l'étape 2 décrit nos contributions aux architectures de QdS conçue en contexte IntServ ;
  - l'étape 3 décrit nos contributions aux architectures de QdS en contexte DiffServ, d'abord mono domaine, puis multi domaines, incluant la signalisation.
- les conclusions générales vis à vis de nos travaux, et notre prospective de recherche font l'objet de la dernière partie.

---

[35] IntServ : Integrated Service Working Group - http://www.ietf.org/html.charters/OLD/intserv-charter.html

[36] DiffServ : Differentiated Service Working Group - http://www.ietf.org/html.charters/OLD/diffserv-charter.html

[37] Et en particulier ceux que nous avions proposés dans l'étape précédente.

[38] DIS : *Distributed Interactive Simulation*

[39] Fonction de contrôle d'admission des connexions (CAC)

[40] *Bandwidth Broker*, tels qu'initialement défini dans [Nich99]

[41] Typiquement des routeurs de bordure

[42] Dans la suite, nous nommerons « chemin de donnée » le chemin des données applicatives.

Plan de l'Etat de l'art

I. Les services et protocoles de Transport à QdS 39

1. Introduction 39

2. Problématique 39

2.1. Evolution du contexte applicatif et réseau 39

2.2. Limites des solutions initiales 40

3. Propositions 41

3.1. Amélioration des mécanismes à des fins de performances 41

3.2. Extensions conceptuelles 43

4. Conclusions 45

II. Les architectures pour la QdS de bout en bout dans l'Internet 46

1. Introduction 46

1.1. Deux modèles précurseurs : IntServ et DiffServ 46

1.2. Problématique générale 48

2. Architectures à QdS basées sur IntServ 49

2.1. Problèmes 49

2.2. Propositions 50

2.3. Conclusions 51

3. Architectures à QdS basées sur DiffServ 52

3.1. Problèmes 52

3.2. Propositions 54

3.3. Conclusions 62

III. La signalisation pour la QdS dans l'Internet multi domaines 63

1. Introduction 63

2. Problèmes 64

2.1. Vis à vis du provisionnement 64

2.2. Vis à vis de la disponibilité des ressources 65

2.3. Vis à vis des modèles à base de BB 65

3. Propositions 66

3.1. L'approche SCCD (signalisation couplée au chemin de donnée) 66

3.2. L'approche SDCD (signalisation découplée du chemin de donnée) 67

4. Conclusion 70

ETAT DE L'ART

Ce chapitre présente l'état de l'art des trois thèmes que nous avons adressés, qui concernent les services et protocoles de Transport (section I), les architectures pour la QdS dans l'Internet (section II), et la signalisation pour l'Internet multi domaines (section III).

I. Les services et protocoles de Transport à QdS

1. Introduction

Le rôle de la couche Transport est d'abstraire les applications des imperfections du réseau sous-jacent. Un protocole de Transport met en œuvre un certain nombre de fonctionnalités[43] concourant à la mise en œuvre du service fourni. Celui ci est accessible par l'application via une interface de programmation, telle que l'API socket, incluant les primitives et paramètres nécessaires à l'invocation du service.

Initialement conçus pour des applications textuelles et des réseaux à bas débit, les protocoles initiaux (UDP et TCP) sont conceptuellement inadaptés au nouveau contexte applicatif et réseau : leurs fonctionnalités ne répondent ni aux besoins des applications multimédia, ni aux évolutions des technologies réseaux support de l'Internet.

Depuis environ 15 ans, de nombreux travaux ont ainsi contribué :

- d'une part, à l'amélioration des performances (débit, délai) des protocoles existants,
- d'autre part, à l'extension des fonctionnalités et des services du niveau Transport, permettant de tirer partie des caractéristiques structurelles des flux multimédia. Ces travaux ont également conduit à reconsidérer l'architecture des protocoles de Transport.

Nos contributions aux services et protocoles de Transport ont abordé le second aspect du problème.

La suite de cette section est structurée comme suit : nous détaillons en section 2 la problématique introduite dans ce paragraphe, puis nous résumons en section 3 les principales propositions adressant les différents aspects de cette problématique.

2. Problématique

2.1. Evolution du contexte applicatif et réseau

A la fin des années 80, les technologies réseaux support de l'Internet étaient peu nombreuses, et offraient de faibles débits d'accès et des taux d'erreurs bits relativement élevés.

Les applications distribuées dans ce contexte étaient uniquement de type client/serveur, manipulaient essentiellement du texte, et ne présentaient que peu ou pas de contraintes sur le débit ou le délai de transit des données échangées.

[43] Destinées, par exemple, à soustraire l'application de la gestion des pertes lorsque le réseau ne peut l'assumer.

Ces dernières années ont vu l'apparition d'un nouveau paysage réseau et de nouvelles applications distribuées.

Les réseaux ont évolué vers les communications à haut débit, puis sans fil :

- les réseaux satellite offrent ainsi de haut débit d'accès mais induisent des délais de transmission beaucoup plus grands que les réseaux filaires ;
- l'explosion des réseaux locaux sans fil et leur exploitation dans la mise en œuvre du concept de réseau *ad-hoc* a encore modifié les caractéristiques du réseau : sa disponibilité ne dépend plus uniquement de la charge utilisateur mais aussi de sa topologie courante.

Les applications sont maintenant multimédia et multi utilisateurs. En cela, elles présentent des caractéristiques très différentes de celles initiales liées, d'une part, à la nature des flux échangés, et d'autre part, au type d'interactivité entre les utilisateurs :

- les nouvelles applications manipulent tous les types d'informations, et notamment l'audio et la vidéo ; en cela :
  - elles présentent un caractère multi flux et un besoin potentiel en synchronisation des flux manipulés (image et son par exemple) ;
  - il existe une redondance dans l'information échangée qui rend certaines pertes acceptables pour l'utilisateur ; il apparaît ainsi un besoin en fiabilité non plus totale mais partielle, la perte (contrôlée) de certaines informations pouvant être tolérées[44] ;
  - elles réclament la disponibilité d'une bande passante faiblement variable pour l'émission des médias continus produits en temps réel ;
- pour les applications classiques, la nature de l'interactivité entre utilisateurs se limite le plus souvent à des échanges sans contrainte sur le délai de transit dans le réseau ; les applications multimédia présentent des types d'interactivité variables :
  - un *streaming* audio/vidéo se traduit par des échanges dans un seul sens mais avec présentation régulière des données en réception ;
  - une visioconférence implique des échanges dans les deux sens et la possibilité que les utilisateurs interagissent comme s'ils étaient à proximité.

  De ce type d'interaction résultent de nouveaux besoins en QdS exprimables en termes de délai de transit borné et éventuellement en termes de gigue[45] bornée.

2.2. Limites des solutions initiales

Les limites des services et protocoles de Transport initiaux s'expriment à deux niveaux :

- au niveau des mécanismes, constituant, dans le nouveau contexte réseau, un goulot d'étranglement vis à vis des performances en débit et en délai ;
- au niveau conceptuel, ce qui induit des fonctionnalités et des services insuffisants au regard des caractéristiques des applications multimédia

Nos travaux ont proposé des solutions aux limites conceptuelles.

---

[44] L'expression de cette contrainte est dépendante de la façon dont est codée l'information. Certains codages (tel que JPEG) n'introduisent aucune dépendance entre les images. D'autres codages (tel qu'H263) introduisent une dépendance entre les images, qui rend indécodables plusieurs images consécutives en cas de perte de certaines d'entre elles, plus importantes.

[45] La gigue désigne la variation moyenne du délai de transit observé.

2.2.1. Limites des mécanismes existants

Les mécanismes des protocoles initiaux présentent deux limites principales :

- les mécanismes classiques de reprise des pertes et de contrôle de flux limitent le débit offert par le réseau ;
- le mécanisme de contrôle des congestions de bout en bout (à base de fenêtre pour TCP[46]), engendre des variations importantes du débit offert aux applications.

2.2.2. Limites conceptuelles

Initialement, les protocoles de Transport sont fondés sur l'alternative conceptuelle suivante :

- soit ils sont *orientés connexion* (tels que TCP) et assurent à leurs utilisateurs un transfert de données sans déséquencements, exempt d'erreurs, de pertes et de duplicatas ; en cela, ces protocoles peuvent être qualifiés de *fiables et ordonnés* ;
- soit ils sont s*ans connexion* (tels que UDP) et ne garantissent ni l'ordre, ni la fiabilité du transfert des données, ces QdS résultant des garanties offertes par le réseau sous jacent ; en cela, ces protocoles peuvent être qualifiés de *non fiables et sans ordre*.

De ce fait, le concept de connexion de type TCP présente deux limites :

- il ne permet pas de prendre en compte au niveau Transport la tolérance (contrôlée) aux pertes, ni les besoins en synchronisation d'un flux multimédia ;
- il ne permet pas de distinguer les différents flux d'une application multimédia : une connexion offre la même QdS à toutes les données soumises par l'utilisateur, sans prendre en compte les besoins spécifiques de chaque flux.

3. Propositions

Face aux limites précédentes, les propositions de solutions ont concouru à :

- l'amélioration des mécanismes existants, pour augmenter les performances ;
- l'extension du concept de connexion, sa mise en œuvre et son évaluation.

3.1. Amélioration des mécanismes à des fins de performances

Chronologiquement, les premières propositions ont adressé le besoin d'améliorer les mécanismes existants, soit dans le cadre des protocoles en place, TCP en l'occurrence, soit dans celui de nouveaux protocoles, tels que XTP[47], puis plus récemment DCCP[48].

3.1.1. La génération XTP

Delta-t [Wats89], VMTP[49] [Cher89] [Nord89] et NETBLT[50] [Clar87] sont les premières propositions faites en ce sens.

---

[46] *Slow start* et *congestion avoidance*

[47] XTP : *Xpress Transfer Protocol*

[48] DCCP : *Datagram Congestion Control Protocol*

[49] VMTP : *Versatile Message Transaction Protocol*

[50] NETBLT : *NETwork BLock Transfer*

Elles ont été suivies par la proposition d'une solution plus innovante, XTP [Ches87] [Ches89], qui symbolisait une nouvelle approche de conception des protocoles, regroupant fonctions de Transport et fonctions de routage.

L'objectif initial du projet était de concevoir un protocole de transfert à haute vitesse, implémentable en circuits VLSI[51] [Schw90].

Les caractéristiques de XTP étaient de deux natures :

- d'une part, il ne définissait pas de service et se présentait comme une boite à outils utilisables, soit par les applications, soit pour construire un service de Transport ;
- d'autre part, certains de ses mécanismes était paramétrables à volonté : l'utilisateur pouvait ainsi activer ou désactiver, par connexion ou par paquet, les mécanismes de contrôle de flux, de détection des erreurs bits et de reprise des pertes.

XTP a aussi introduit la notion de vecteur d'acquittements sélectifs, permettant au récepteur de mentionner les pertes observées dans le flux de données reçues. En complément du contrôle de flux par fenêtre, XTP a introduit la notion de contrôle de débit par rafale.

3.1.2. Evolutions des mécanismes de TCP

L'implémentation d'XTP n'ayant pas abouti, les travaux visant à améliorer les performances des protocoles de Transport ont essentiellement visé TCP :

- le contrôle de flux à base de fenêtre propre à TCP a été reconsidéré (extension de la taille de la fenêtre) pour répondre aux ruptures de débit engendrées dans le contexte des réseaux caractérisés par un produit « délai × débit » important, cas typique des réseaux satellite [Podu98] ;
- le contrôle des pertes initial de TCP[52] a été étendu pour intégrer la notion de vecteur d'acquittement [Math96] [Fall96], l'objectif étant d'éviter que soient retransmises des données déjà reçues et stockées en réception ;
- le contrôle des congestions du réseau, initialement à base de fenêtre, a été reconsidéré pour induire des variations moindres du débit offert au niveau applicatif [Hand03].

3.1.3. DCCP

Le besoin d'un contrôle de congestion n'est pas spécifique à TCP, aussi en retrouve-t-on les propositions d'extension dans certains nouveaux protocoles, notamment DCCP [Kohl02].

Le contrôle de congestion de bout en bout est indispensable dans l'Internet pour éviter un effondrement du réseau [Floy99]. Actuellement, les applications multimédia basées sur UDP ont l'obligation d'implanter un tel contrôle à leur niveau (UDP ne le faisant pas) de façon, d'une part à ne pas surcharger le réseau en cas de congestion, et d'autre part, à assurer une certaine équité vis à vis des flux TCP dans l'utilisation de la bande passante.

DDCP est une proposition de protocole orienté connexion mais sans garantie de fiabilité, offrant notamment comme possibilité à son utilisateur de sélectionner le contrôle de congestion le plus adapté à ses besoins.

---

[51] Le projet *Protocol Engines Incorporated* a finalement été arrêté faute d'une implémentation suffisamment performante.

[52] Basé sur des acquittements cumulatifs.

3.2. Extensions conceptuelles

Les propositions précédentes présentent cependant des limites conceptuelles vis à vis de leur adéquation aux besoins du multimédia.

Ces limites sont de deux ordres :

- la première est relative à l'absence de considération de la tolérance aux pertes et du besoin en synchronisation des flux multimédia ;
  - TCP ignore la tolérance aux pertes et aux erreurs des médias continus, et ne considère pas non plus leur besoin potentiel en synchronisation ; entre outre, il assure une fiabilité totale au coût d'un accroissement non contrôlé du délai de transit et du volume transféré ;
  - à la différence de TCP, XTP introduit la possibilité de construire un service de Transport garantissant une fiabilité non plus totale mais partielle, sur la base de mécanismes de gestion des pertes paramétrables. Cependant, son mode de service orienté flux d'octets (comme TCP) n'est pas adapté à ce type de gestion [Chas95] ;
  - l'objectif premier de DCCP est d'offrir aux applications un service en mode message à contrôle de congestion paramétrable ; cependant, il ne considère ni les besoins en fiabilité partielle contrôlée (le service est non fiable), ni les besoins en synchronisation ;
- la deuxième limite est liée à l'absence de prise en compte du caractère multi flux des applications, présentant chacun des besoins spécifiques en QdS.

Nous avons proposé une solution à ces deux limites en contribuant :

- d'une part, à l'élaboration d'une nouvelle famille de protocoles, *à ordre et fiabilité partiels* ;
- d'autre part, à la définition d'une nouvelle architecture de protocole, *à multi connexions*.

3.2.1. Le concept de connexion à ordre et fiabilité partiels (POC)

Les protocoles de Transport initiaux sont fondés sur une alternative conceptuelle : soit ils sont *avec connexion* (tels que TCP) et offrent des garanties d'ordre total et de fiabilité totale sur le transfert des données utilisateur, soit ils sont *sans connexion* (tels que UDP) et ne garantissent ni ordre ni fiabilité.

Cette classification fait apparaître un vide conceptuel entre les deux familles de protocoles, suggérant une extension du concept de connexion, vis à vis de l'ordre et de fiabilité. Ce nouveau concept, la *connexion à ordre et fiabilité partiels* (POC), a été introduit dans nos travaux [Amer94] [Diaz94] [Chas95].

Une POC est une connexion de Transport permettant de définir et de mettre en œuvre tous les services d'ordre partiel et de fiabilité partielle entre deux entités communicantes.

Il étend et unifie les deux approches classiques de la notion de connexion représentées par UDP et TCP, qui apparaissent comme deux cas extrêmes de cette proposition[53]. Ces travaux sont résumés dans l'Etape 1 de cette partie du mémoire.

De nombreux travaux ont contribué à la mise en œuvre et à l'évaluation du concept de POC. Nous les présentons ci-après.

[53] Illustratif des protocoles avec connexion, ordonnés et non fiables, DCCP apparaît aussi comme un cas particulier de POC.

3.2.2. Mise en œuvre et évaluation du concept de POC

L'évaluation du concept a été abordée sous trois aspects, analytique, en simulation et expérimental en contexte réel :

- [Four97b] propose une évaluation des performances d'un modèle OPNET du protocole POC. Il analyse l'impact de différents choix de QdS (ordre et fiabilité) sur les ressources mémoires utilisées et sur les performances (délai de transit) obtenues. Il prend en compte différents types de réseaux caractérisés par des taux de pertes et des RTT différents ;
- [Mara96] complètent ces résultats par un modèle analytique visant à démontrer les gains induits par un service partiellement fiable, comparativement à TCP. [Mara97] valide les résultats précédents en les confrontant à un modèle développé en simulation (OPNET).

La mise en œuvre du concept de POC a été étudiée de multiples façons :

- [Cono94] propose des modifications à apporter à TCP pour y introduire une gestion de l'ordre et de la fiabilité partiels ;
- [Roja99] définit un protocole à fiabilité partielle pour les flux vidéo MPEG. Il tire partie de la structure des flux MPEG en images I, B, P pour améliorer la QdS d'une application de *streaming* vidéo. [Lecu01] cible le même objectif pour une application interactive ;
- [Ihid02] propose une instanciation du concept de fiabilité partielle par codes de *Reed Solomon* pour les flux vidéo MPEG distribués en temps réels ;
- [Expo03] [Expo04] définissent un cadre de spécification UML 2.0 de protocole de Transport configurable pour la QdS. Sur ces bases, il développe un protocole à fiabilité partielle, FPTP[54], qui tient compte des contraintes temporelles de l'utilisateur et inclut un contrôle de congestion orienté QdS de type TFRC.

3.2.3. Les protocoles à multi connexions

Les propositions précédentes ont exploité le concept de POC de deux façons :

- soit sans tenir compte de la structure des flux multimédia : toutes les données émises se voient offrir la même QdS[55], en cela, ces protocoles sont *mono connexion* et *monomédia* ;
- soit en cherchant à différencier le traitement des objets d'un même flux (images I, B, P des flux H263/MPEG par exemple) ; en cela, ces protocoles sont *mono connexion* et *multimédia*.

Cependant, aucune des deux familles ne permet de gérer une quelconque forme de synchronisation entre flux différents d'une même application.

Face à ce besoin, une nouvelle architecture de protocoles, *à multi connexions*, a été proposé dans nos travaux [Chas96]. Cette architecture (MM-POC[56]) permet l'instanciation puis la coordination de plusieurs POC offrant chacune une QdS spécifique au flux véhiculé.

La gestion de l'ordre et de la fiabilité se situe à deux niveaux, dans chaque connexion et entre les connexions. Cette architecture, présentée plus en détails dans l'Etape 1, permet de prendre en compte et de tirer partie des contraintes de fiabilité et de synchronisation logique à la fois intra et inter-flux.

---

54 FPTP : *Fully Programmable Transport Protocol*

55 Respect d'un taux maximal de perte et/ou d'un nombre maximal de pertes consécutives par exemple.

56 MM-POC : *Multimedia Partial Order Connection*

Cette proposition a été exploitée de deux façons :

- [Four96] propose une implémentation du protocole MM-POC, sous forme de module STREAM ; son modèle de représentation de l'ordre partiel est une liste d'entiers représentant les successeurs immédiats de chaque objet. La fiabilité partielle est représentée par un taux maximal de pertes et un nombre maximal de pertes consécutives ;
- [Four97c] propose une évaluation comparative des performances de modèles (OPNET) des protocoles POC et MM-POC ;
- dans sa proposition d'architecture pour garantir les contraintes de synchronisation temporelle d'une visioconférence, [Owez98] tire partie d'un Transport MM-MOC pour minimiser le taux de pertes observé au niveau applicatif, comparativement à une utilisation de UDP. Un modèle basé sur les réseaux de Petri est utilisé pour que l'application exprime ses besoins au service de Transport sous jacent ;
- [Bert02] instancie l'architecture MM-POC dans un contexte de multi réseaux hétérogènes (ATM, satellite, …), d'une part au niveau de l'utilisateur qui dispose de plusieurs accès réseau, et d'autre part sur le chemin de donnée. Il base en particulier sa proposition sur une rupture des connexions Transport par des proxys situés en bordure de chaque réseau, et montre les gains induits par une gestion de l'ordre et de la fiabilité partiels.

Indépendamment de nos travaux, le principe d'une architecture à multi connexions a également été adopté et implanté dans la proposition SCTP [Stew00], soutenue à l'IETF.

SCTP n'est pas un protocole initialement conçu pour le transfert d'applications multimédia. Il a été développé pour le transfert de la signalisation dans le cadre d'application de voix sur IP. Cependant, il est appelé à couvrir un spectre plus large de besoins applicatifs.

SCTP est un protocole orienté connexion offrant un service en mode message[57] :

- il permet à ses utilisateurs d'établir des *associations* composées chacune de plusieurs connexions de type TCP, sans contrainte d'ordre entre ces connexions. L'objectif initial est de permettre la délivrance des données d'une connexion, indépendamment de l'état des autres connexions. Notons que la notion d'association SCTP correspond à un cas particulier de multi connexions à ordre et fiabilité partiels[58] ;
- il intègre également le support du *multi homing* (un point d'accès au service peut être adressé par plusieurs adresses IP, rattachées chacune à une interface physique différente). L'objectif initial était d'introduire une redondance, permettant de remédier au problème de panne du réseau. La balance de charge en vue d'améliorer la QdS est une opportunité ouverte. Notons que cette possibilité rejoint les propositions faites dans [Bert02].

4. Conclusions

Le niveau Transport a été très étudié à une époque où il n'était pas clair que se dessinerait la volonté d'introduire une QdS de niveau IP dans l'Internet.

Face aux limites des protocoles existants, plusieurs approches ont été adoptées pour :

- améliorer les performances, notamment vis à vis du débit, des protocoles existants,
- étendre les fonctionnalités de ces protocoles.

---

[57] Comme UDP et à la différence de TCP qui offre un service en mode flux d'octets.

[58] Ordre total et fiabilité totale sur chaque connexion et ordre nul entre les connexions.

C'est cette deuxième voie que nous avons suivie. Nos solutions permettent d'optimiser l'utilisation des ressources du réseau (en particulier sa bande passante), et de minimiser le délai de transit des communications tout en garantissant les contraintes minimales de l'application à en termes d'ordre et fiabilité.

Les promesses des travaux récents menés pour garantir une QdS IP dans l'Internet pourraient occulter l'intérêt d'un Transport à QdS. Cependant, plusieurs raisons motivent la poursuite des travaux sur ce thème :

- d'abord, la QdS dans l'Internet multi domaines restera toujours dépendante du mode d'administration de chaque domaine : il existera donc toujours des portions de l'Internet inaptes à offrir des garanties de QdS au niveau IP ;
- ensuite, lorsqu'elle sera effectivement garantie, cette QdS aura un coût, et les solutions d'optimisation de l'utilisation de la bande passante auront un intérêt dans ce nouveau contexte ; nos travaux d'intégration des niveaux Application/Transport/IP DiffServ ciblent en particulier cet objectif ;
- enfin, l'explosion des technologies sans fil et l'émergence de nouveaux concepts de réseaux, notamment mobiles et ad hoc, nécessiteront de reconsidérer la notion de garantie ; il s'agira dans certains cas, de gérer la pénurie de bande passante, et le besoin d'optimiser la qualité des communications se posera donc dans un contexte renouvelé.

Nous présentons maintenant un état de l'art des propositions d'architectures pour la garantie de QdS dans l'Internet, incluant de nouveaux services au niveau IP.

II. Les architectures pour la QdS de bout en bout dans l'Internet

1. Introduction

L'objectif d'offrir d'autres services de niveau IP au delà du seul *Best Effort* impose une remise en cause du modèle de service des paquets IP et du modèle d'architecture de l'Internet :

- alors qu'un routeur IP classique sert les paquets selon une politique de type FIFO et *Tail Drop*[59], de nouvelles fonctions sont nécessaires pour permettre aux routeurs de distinguer les paquets IP, de les ordonnancer différemment en sortie, ou encore de retarder/écarter les trafics moins prioritaires ;
- pour obtenir des garanties de QdS, il faut satisfaire des besoins supplémentaires, tels que la disponibilité des ressources sur le chemin de donnée. Ceci nécessite des fonctions de contrôle et/ou de gestion du réseau, telles que le contrôle d'admission des requêtes ou les fonctions relatives à la planification du réseau ;
- enfin, certaines des fonctions requises nécessitent l'échange d'informations entre les équipements du réseau, avant ou durant le transfert des données. Ces échanges peuvent être effectués au travers des paquets de données, ou via des protocoles de *signalisation*.

1.1. Deux modèles précurseurs : IntServ et DiffServ

Deux groupes de travail de l'IETF, IntServ puis DiffServ, ont été successivement constitués au milieu des années 90 pour étudier la mise en œuvre de nouveaux services au niveau IP.

[59] Le *Tail Drop* consiste à rejeter les paquets arrivant dans une file d'attente pleine, sans prise en compte de priorité.

1.1.1. IntServ

Le modèle IntServ introduit deux nouveaux services IP, GS [Shen97] et CL[60] [Wroc97]. GS veut satisfaire les besoins des applications à fortes contraintes temporelles ; CL vise à répondre aux applications moins exigeantes, et notamment les applications adaptatives.

IntServ repose sur trois principes fondamentaux :

- une réservation de ressources est effectuée par *session* au niveau de chacun des hôtes et routeurs du chemin de donnée ; elle se traduit par l'établissement et le maintien d'un état ; une session désigne au moins un socket de destination (unicast ou multicast) ; la granularité la plus fine est le flux ;
- cette réservation est effectuée via protocole RSVP[61] [Brad97a] [Brad97b], dans le cadre d'une phase d'établissement de réservation ; celle-ci est initiée par le(les) émetteur(s) mais ce sont le(les) récepteurs qui effectuent la requête de réservation. La requête est propagée de routeur en routeur, qui effectuent chacun tour à tour un contrôle d'admission local[62], et enregistrent en cas de succès un état de réservation pour la session considérée ;
- le maintien des réservations est assuré par la réception de messages de rafraîchissement, émis par le(les) émetteurs. Une réservation peut également être relâchée explicitement.

Le modèle IntServ présente des limites importantes qui en interdisent le déploiement à l'échelle du plein Internet :

- d'une part, le principe de réservation au niveau de tous les routeurs, qui nécessite un état par flux et la signalisation nécessaire à son maintien ; se pose alors un problème de mise à l'échelle de l'approche vis à vis du nombre de flux à QdS ;
- d'autre part, l'ignorance de la structure multi domaines de l'Internet ; cette structure de fait reflète la nécessité d'administrer les domaines de façon indépendante vis à vis de leurs politiques internes de gestion du routage, de la QdS ou de la sécurité.

1.1.2. DiffServ

Le modèle DiffServ [Blak98] a été conçu pour répondre aux limites d'IntServ. Il est basé sur les principes fondamentaux suivants :

- DiffServ écarte la notion de flux en cœur de réseau, et introduit à la place celle de *classe*[63] : les paquets d'une même classe sont traités identiquement par tous les routeurs du chemin empruntés entre l'entrée et la sortie du domaine considéré ;
- à la différence d'IntServ, DiffServ ne propose pas de contrôle d'admission[64], mais s'appuie sur la notion de contrat *(SLA[65])* négocié entre utilisateur et fournisseur, préalablement au transfert des données. La partie technique d'un SLA, le SLS[66], spécifie (à minima) la quantité maximale de trafic de chaque classe que l'utilisateur a le droit d'injecter dans le domaine, ainsi que le traitement (*conditionnement*) que subiront les paquets entrants ;

---

60 GS: *Guaranteed Service* et CL: *Controlled Load*

61 RSVP : *ReSource reserVation Protocol*

62 Fonction de la technologie réseau sous jacente.

63 Paquets IP identifiés par une même marque explicite (champs TOS des paquets IPv4 ou *Traffic Class* des paquets IPv6)

64 Aucun protocole de signalisation n'est en conséquence défini à cette fin.

65 SLA : *Service Level Agreement*

66 SLS : *Service Level Specification*.

– pour une classe de service, un SLA garantit un comportement (*PHB*[67]) identique sur chaque routeur, à l'échelle du domaine ; deux PHB ont été standardisés, EF [Jaco99] et AF[68] [Hein99]. EF vise un service équivalent à celui d'une ligne dédiée, AF vise un service à débit minimal garanti, avec différents niveaux de priorité à la perte et au délai ;
– en entrée du réseau, le conditionnement des paquets se traduit d'abord par leur classification, sur la base soit d'une marque unique[69], soit de plusieurs champs d'entête pouvant porter sur les données utilisateur[70]. S'en suit un contrôle de la conformité du trafic au regard du SLS, et l'application de mesures de mise au gabarit (*policing*) pour le trafic non conforme ; ce *policing* peut se traduire par le rejet, le retard ou le basculement dans une autre classe. Le trafic conforme est marqué suivant sa classe d'appartenance, puis injecté dans le réseau. L'ordonnancement[71] des paquets dépend de leur classe.

1.2. Problématique générale

Chacun des modèles est confronté aux deux mêmes problèmes :

– le premier est relatif à la délivrance du service de bout en bout, qui résulte :
  • de la disponibilité effective des ressources sur le chemin de donnée ;
    – IntServ l'assure ;
    – DiffServ laisse le problème ouvert ;
  • du provisionnement du service ;
    – pour IntServ, ceci consiste à configurer, pour chaque routeur, les fonctions de traitement des paquets, pour obtenir les services définis ;
    – pour DiffServ, le problème est plus complexe car il inclut aussi le dimensionnement adéquat des ressources du domaine[72] au regard des SLS souscrits ;
– le deuxième problème concerne la signalisation, nécessaire, par exemple, pour s'assurer de la disponibilité des ressources tout au long du chemin de donnée ;
  • IntServ s'appuie sur RSVP ;
  • DiffServ laisse le problème ouvert.

Pour DiffServ, le passage au multi domaines accroît toutes les difficultés du fait :

– de la présence de liens inter domaines, dont il s'agit également d'assurer la disponibilité sur le chemin de donnée ;
– de l'hétérogénéité des domaines vis à vis des classes de services supportées et de leurs performances ; il s'agit donc de définir comment provisionner le service de bout en bout.

Nous avons adressé cette problématique générale, d'abord dans le contexte IntServ et ensuite dans le contexte DiffServ, en cherchant à répondre au problème supplémentaire de l'utilisateur, qui se traduit en ces termes : *comment assurer une concaténation de services IP non définis en termes de performance, qui, en tenant compte de l'impact du Transport, satisfasse les besoins applicatifs à un coût minimal en termes de ressources*.

---

[67] PHB : *Per Hop Behaviour*

[68] EF: *Expedited Forwarding*, AF: *Assured Forwarding*.

[69] Champs TOS des paquets IPv4 par exemple.

[70] Typiquement l'entête des PDU de niveau Transport (n° de port source et/ou destination).

[71] *Priority Queuing*, *Weighted Fair Queuing*, …

[72] Topologie des routeurs, dimensionnement des liens, …

Ce dernier problème se pose comme suit.

▪ Pour utiliser les nouveaux services IP, il est nécessaire de caractériser le profil du trafic applicatif[73]. Si pour les premières applications multimédia, cette caractérisation paraissait relativement simple, leur évolution a conduit à des modèles plus complexes résultant :

– de la sémantique des données échangées (nouveaux codecs pour l'audio et la vidéo, redondance d'information pour la simulation distribuée, …) ;
– de l'interactivité entre les participants (suppression des blancs en téléphonie, taux de rafraîchissement variables des informations d'état dans une simulation distribuée, …).

▪ En conséquence, il apparait nécessaire :

– de distinguer dans le trafic applicatif les différentes composantes, en les classifiant d'après leurs besoins en QdS ;
– et pour chacune de ces composantes :
  • de prédire et de caractériser le profil de trafic associé ;
  • de choisir les services Transport et IP les plus appropriés vis à vis de leurs besoins.

En adoptant un point de vue utilisateur, nos propositions d'architectures pour la QdS dans l'Internet intègrent des propositions de solution à ce problème, visant :

– à abstraire aux programmeurs applications, la connaissance, le choix et le paramétrage des protocoles de niveaux Transport et IP sous jacents ;
– à minimiser l'utilisation des ressources réseaux les plus coûteuses, ce qui présente un intérêt, d'une part pour l'utilisateur, et d'autre part pour le fournisseur.

La suite de cette section est structurée de la façon suivante. Le paragraphe (2) présente les architectures à QdS basées sur IntServ. Celles basées sur DiffServ sont présentées dans le paragraphe (3).

2. Architectures à QdS basées sur IntServ

Ce paragraphe introduit les deux problèmes majeurs posés par le modèle IntServ, et présente les principaux types de réponses à celui que nous avons adressé.

Cette partie de l'état de l'art est plus courte que la suivante pour deux raisons :

– d'une part, IntServ répond en grande partie aux problèmes d'architectures internes au réseau ; il ne reste à définir que la façon d'en utiliser les services ;
– d'autre part, les limites de l'approche à l'échelle de l'Internet ont rapidement conduit à ce que les travaux se focalisent sur le modèle DiffServ.

Notons que d'autres travaux, non présentés dans ce mémoire, ont contribué à définir comment permettre l'inter fonctionnement des deux approches, IntServ étant perçue, généralement, comme une solution de bordure.

2.1. Problèmes

IntServ répond en grande partie aux problèmes de provisionnement et de disponibilité des ressources sur le chemin de donnée.

[73] Débit, taille des unités de données applicatives, …

Pour cela, il impose une solution basée sur la réservation de ressources par flux[74], au niveau de chaque routeur du chemin de donnée, et sur un contrôle d'admission local à chaque routeur dont la signalisation s'appuie sur protocole RSVP [Brad97a] [Brad97b].

IntServ ne définit pas comment implanter les fonctions de traitement des paquets au niveau des routeurs. L'implantation du contrôle d'admission local à chaque routeur est fonction de la technologie réseau sous jacente.

Les principaux problèmes posés par IntServ sont de deux ordres :

- le premier concerne sa résistance au facteur d'échelle, vis à vis du nombre de flux ; il s'agit ici de définir comment minimiser, d'une part, le nombre d'états de réservation qu'un routeur est susceptible de maintenir, et d'autre part, le nombre de messages de signalisation nécessaires au maintien de ces états par rafraîchissement périodique ;
- le deuxième problème consiste à définir l'architecture de bout en bout permettant d'interfacer les applications avec les services IntServ.

Nos contributions ont adressé ce second problème pour une famille spécifique d'applications, les applications de simulation interactive distribuées (DIS). Les propositions décrites ci-après illustrent les principales réponses.

2.2. Propositions

Plusieurs propositions ont adressé le besoin d'interfacer les applications avec les solutions de gestion de la QdS au niveau réseau, en particulier suivant l'approche IntServ.

Pour IntServ, l'API RSVP [Brad98] a initialement été conçue à ces fins ; elle permet à son utilisateur de préciser le profil du trafic sur lequel doit porter la réservation, ainsi que le service requis (GS ou CL). L'API Winsock2 développée par Microsoft, visait en outre l'accès à d'autres types de services, notamment ceux envisagés pour un environnement tout ATM.

L'API RSVP présente deux difficultés :

- la première est qu'il est impossible d'établir une réservation depuis des machines autres que celles sur lesquelles s'exécute l'application ; cette limite résulte de l'approche IntServ, qui impose que les hôtes émetteur(s) et récepteur(s) soient inclus dans la réservation ; elle est actuellement reconsidérée par les travaux du groupe de l'IETF NSIS[75] ;
- la deuxième difficulté, que nous avons adressée, provient du fait que les paramètres de spécification du trafic sont clairement orientés réseau[76]. Une mauvaise spécification engendre soit une sur-allocation de ressources, soit un écartement des paquets qui ne respectent pas le profil annoncé.

Face à cette seconde difficulté, plusieurs propositions, dont la notre, visent à abstraire à l'application le choix et le paramétrage des services IntServ ; il s'agit ici d'offrir une interface dont les paramètres ont une sémantique simple du point de vue du programmeur.

[Yan01] étend l'API RSVP, de façon à prendre en compte différents types de trafic applicatif (vidéo MPEG, audio, …) et différents niveaux de réservation (fort, moyen, faible). Il fournit un ensemble d'appels de librairies *ad-hoc* développées au dessus de l'API RSVP.

---

[74] En toute rigueur : au moins par socket destination.

[75] Que nous présenterons en section IV.

[76] Débit moyen/crête en octets de paquets IP par seconde, taille des rafales en octets de paquets IP, …

Nos travaux ciblent le même objectif pour une famille d'application (DIS) illustrative des jeux distribués, mais explicite aussi la décomposition du trafic afin d'optimiser l'utilisation des ressources du réseau [Chas99] [Chas00]. En outre, l'application est déchargée du choix des services IP. L'API proposée[77] comporte des paramètres décrivant le trafic du point de vue de l'utilisateur[78], qu'elle traduit en paramètres réseau, avant d'invoquer l'API RSVP.

[Alme97] [Jesp97] adressent le même objectif d'abstraction, mais sur la base de solutions réseau différentes d'IntServ, en particulier ATM (en natif). Certains étendent leurs solutions à un contexte IntServ/RSVP sur ATM [Eber98], puis sur d'autres technologies réseaux (FDDI notamment). Ceci les conduit à définir comment convertir les paramètres et services IntServ en ceux de la technologie réseau considérée.

Dans le même ordre d'idée, d'autres travaux proposent d'abstraire le programmeur d'applications du type de services offerts au niveau réseau (IntServ, DiffServ et ATM natif) [Shieh01] [Shih04]. Suivant cet objectif, [Wang00] propose par exemple une couche de contrôle de ressources (RCL) de niveau middleware permettant à l'application de spécifier ses besoins en QdS de deux façons, qualitative ou quantitative. RCL comporte trois niveaux : le premier niveau interface les applications avec le système proposé, directement ou indirectement[79] ; le second niveau sélectionne le service réseau, génère les paramètres correspondants puis invoque son activation ; le service est mis en œuvre dans le troisième niveau (RSVP pour IntServ par exemple).

2.3. Conclusions

IntServ est le premier modèle qui a été proposé pour la QdS au niveau IP.

C'est celui que nous avons d'abord considéré dans nos travaux, en nous fixant un double objectif : d'une part, abstraire au niveau applicatif la complexité d'utilisation des services IntServ, et d'autre part, optimiser l'utilisation des ressources du réseau, en considérant l'impact du Transport.

Nos n'avons contribué à la deuxième partie de cet objectif que dans le contexte DiffServ pour les raisons exposées ci-après.

IntServ présente des limites importantes inhérentes à son principe de base : une réservation par session sur tous les équipements (hôtes et routeurs) du chemin de donnée. En cela, IntServ ignore la structure multi domaines de l'Internet en imposant que tous ses acteurs adoptent un seul et même principe de gestion de la QdS. En outre, il est aujourd'hui unanimement admis que cette approche, à l'échelle du plein Internet, n'est pas viable vis à vis du nombre potentiel de flux à QdS traversant un routeur.

Dans ce contexte, IntServ a rapidement fait place à l'approche DiffServ. Par son principe de différenciation de trafic par classe en cœur de réseau, et par la liberté de gestion intra domaine qu'il laisse aux opérateurs de l'Internet, DiffServ est aujourd'hui reconnu comme l'unique cadre pour obtenir des garanties de QdS à l'échelle de l'Internet multi domaines.

Nous présentons donc à présent les propositions d'architectures à QdS dans ce contexte.

---

[77] Accessible par l'application ou via une IHM (Interface Homme Machine).

[78] Tel que le nombre et le type des entités (char, avion, …) impliquées localement.

[79] Par le biais d'un programme annexe.

3. Architectures à QdS basées sur DiffServ

Le modèle DiffServ, proposé pour pallier aux limites d'IntServ, laisse ouverts des problèmes importants vis à vis de l'obtention d'un service IP à QdS de bout en bout, pour lesquels les solutions à l'échelle du multi domaines sont encore à ce jour insatisfaisantes.

Nous les présentons ci-après, puis nous exposons ensuite les propositions de solutions.

3.1. Problèmes

Les deux problèmes majeurs laissés ouverts par DiffServ sont :

- le provisionnement des services au regard des SLS ;
- la disponibilité des ressources sur le chemin de donnée.

Le passage au multi domaines accroît la complexité des réponses, tant vis à vis de la délivrance du service que de la signalisation nécessaire.

Il faut en effet offrir à l'utilisateur des assurances analogues, que les données soient échangées à l'intérieur du domaine ou bien qu'elles aient à emprunter d'autres domaines pour aboutir au destinataire.

Les solutions sont d'autant plus complexes que l'hétérogénéité des domaines est grande[80].

Idéalement, tous les domaines déploient tout ou partie des mêmes classes de services et appliquent le modèle retenu pour la gestion de la QdS multi domaines[81].

Cependant, l'Internet du futur sera certainement différent. Quatre types de domaines pourraient être distingués :

- certains domaines offriront des services différenciés, et :
  - soit appliqueront le modèle de QdS retenu pour le multi domaines (type 1) ;
  - soit ne l'appliqueront pas : pour chaque classe, le trafic entrant, à concurrence de la quantité spécifiée dans le contrat, obtiendra les performances du PHB correspondant jusqu'à son point de sortie ; cependant, aucune garantie ne sera fournie quant à la continuité du service sur le lien de sortie (type 2) ;
- certains domaines n'offriront pas de services différenciés et :
  - soit n'offriront aucune garantie de QdS (type 3) ;
  - soit seront sur provisionnés en bande passante : le trafic entrant, à concurrence de la bande passante offerte sur le lien d'entrée, ressortira du domaine avec des performances analogues à celles d'une ligne dédiée entre l'entrée et la sortie ; aucune garantie ne sera fournie quant à la continuité du service en sortie du domaine (type 4)[82].

Jusqu'à présent, toutes les propositions de gestion de la QdS (dont la notre) ont été formulées dans le contexte d'un Internet multi domaines homogène du type 1. Parmi les grands projets, seul EuQoS vise à prend en compte une totale hétérogénéité dans sa seconde phase[83].

---

[80] Il en va de même pour de même pour les solutions de signalisation.

[81] En intra domaine, l'administrateur est libre d'appliquer ses propres règles.

[82] Ce cas peut être perçu comme un cas particulier du type 2.

[83] Prévue à partir de mars 2006 en cas de continuation du projet suite à sa première phase

Nous détaillons à présent les problèmes du provisionnement et de la disponibilité des ressources dans le contexte du multi domaines.

3.1.1. Le problème du provisionnement des services

Le provisionnement d'un service consiste, pour le gestionnaire du domaine, à accorder le dimensionnement des ressources offertes par le réseau aux besoins potentiels des clients (Figure 4) :

- ces clients peuvent être des particuliers, des entreprises ou bien d'autres fournisseurs ;
- les accords entre domaines adjacents sont qualifiés d'*accords de peering*. Actuellement, ces accords sont négociés pour des durées assez longues ; l'introduction d'une dynamicité dans ces accords reste aujourd'hui un problème ouvert.

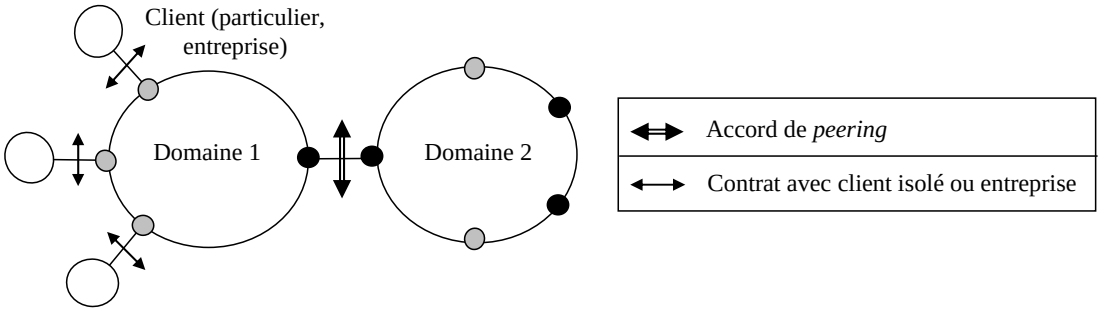


Figure 4 : Les différents types de contrats

On considère qu'un service est bien provisionné si son fournisseur est capable d'assurer le service depuis le point de souscription du contrat (SLA) jusqu'au(x) point(s) de sortie défini(s) dans le contrat, en respectant les performances annoncées[84].

Le problème du provisionnement comporte deux parties :

- la première, que nous n'avons pas abordée, s'exprime en termes de choix de topologie, de dimensionnement des liens et d'ingénierie de trafic au sein de chaque domaine ;
- la deuxième partie du problème, que nous avons traitée, est inhérente à la nature multi domaines de l'Internet. Elle part du fait que les données émises par les clients vont au delà de leur domaine d'accès, et résulte du constat que les domaines traversés (même s'ils sont homogènes) n'offrent ni les mêmes capacités, ni les mêmes performances pour leurs classes de service. Il est donc nécessaire de connaitre les classes de services (et leurs caractéristiques) des domaines impliqués dans le chemin de donnée pour effectuer des choix de concaténation répondant aux contrats de service souscrits par les clients.

En outre, et pour les deux aspects du problème, le point de vue fournisseur conduit à trouver un compromis entre le dimensionnement des ressources (incluant les accords de *peering* en sortie) et la satisfaction de service assurée aux contractants.

3.1.2. Le problème de disponibilité des ressources

Le provisionnement ne suffit pas à garantir à tout instant la disponibilité des ressources sur le chemin des données.

---

[84] Notons qu'actuellement et dans les faits, ces points de sortie ne sont pas mentionnés dans les contrats, auquel cas, nous (clients), sommes en droit d'espérer que la portée de celui ci soit au delà du domaine de souscription …

En environnement mono domaine, ceci résulte du compromis adopté par le fournisseur entre le dimensionnement des ressources et la satisfaction de la demande globale de ses clients :

- si ce compromis se traduit par un sur dimensionnement des ressources internes du domaine, le problème ne se pose pas ;
- sinon, sur certains liens, des ressources peuvent être temporairement indisponibles.

Pour le multi domaines, le problème se pose car le sur dimensionnement n'est pas possible à l'échelle du multi domaines. En effet, il engendrerait par récursivité une explosion des accords de *peering*. La Figure 5 donne un exemple illustratif en soulignant le problème pour l'accord entre les domaines 5 et 6, dans le sens 5 vers 6.

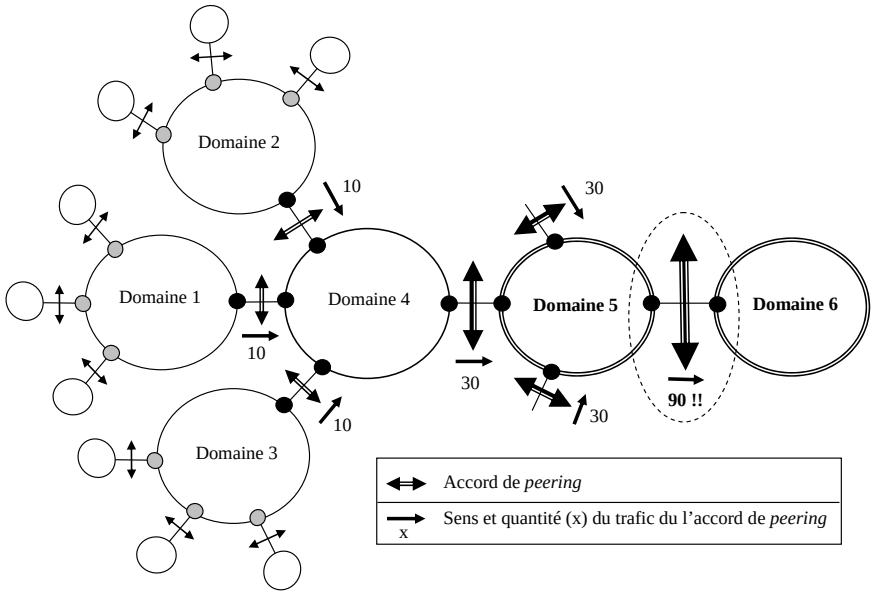


Figure 5 : Problème de sur dimensionnement à l'échelle du multi domaines

La solution repose sur un contrôle d'admission au moment des requêtes de service (CAC[85]) formulées par les clients dans le cadre de leur contrat. L'objet de ce contrôle est de statuer sur la disponibilité des ressources sur chacun des domaines traversés, mais également sur les liens reliant ces domaines. En fonction de la réponse, la requête est acceptée ou rejetée.

Enfin, pour toutes les requêtes acceptées, l'engagement du fournisseur de service est de protéger, à l'intérieur de chaque agrégat, le trafic conforme vis à vis du trafic non conforme.

Dans cette problématique, nous avons supposé le CAC intra domaine résolu et disponible.

3.2. Propositions

De nombreux travaux ont traité du provisionnement et de la disponibilité des ressources, soit de façon isolée, soit dans le cadre d'architectures, souvent issues de grands projets.

Nous illustrons ici ces travaux au travers des cadres architecturaux proposés, en positionnant nos contributions dans cet état de l'art.

85 CAC : *Connection Admission Control*

3.2.1. Introduction

Plusieurs initiatives (Internet 2[86] aux Etats Unis, NGNI[87] en Europe) ont été menées depuis six ans conduisant à l'élaboration de grands projets :

- QBone[88], lancé en 1999 aux Etats Unis, est un projet précurseur destiné à la mise en œuvre d'un service PREMIUM[89] à l'échelle de l'Internet multi domaines ; le projet n'a pas abouti au résultat initialement ciblé, et a ensuite visé un objectif de service moins performant ne requérant ni réservation ni contrôle d'admission ;
- au niveau européen, trois projets menés conjointement, AQUILA[90], TEQUILA[91] et CADENUS[92] ont proposé les premières architectures fonctionnelles dans l'objectif commun d'offrir de la QdS de niveau IP dans l'Internet multi domaines [Gior03] ;
- plus récemment, le projet MESCAL[93] a étendu les résultats des projets CADENUS et TEQUILA, en étudiant plus spécifiquement le problème du provisionnement à l'échelle du multi domaines ; sa proposition d'architecture est basée sur un nouveau concept de contrat client/fournisseur et d'accords de *peering ;*
- sur la base de l'ensemble de ces contributions, le plus récent de ces projets, EuQoS[94] (actuellement en cours et auquel nous participons), propose et développe une architecture de gestion de la QdS pour l'Internet multi domaines hétérogènes ; il place son effort sur le contrôle d'admission et la signalisation en envisageant le provisionnement sur la base d'accords de *peering* plus simples que ceux définis dans TEQUILA et MESCAL.

A l'heure actuelle, aucune des solutions proposées ne répond au problème de la garantie de QdS dans sa globalité, ni même ne s'impose comme la référence pour la suite des études. Ceci tient d'une part à la complexité du problème dans son ensemble, et d'autre part à l'absence de consensus dans la conceptualisation de ses différents aspects. La *résistance au facteur d'échelle* et la notion de *contrat* en constituent deux exemples :

- le maintien des réservations par flux sur chaque routeur (à la mode IntServ) est unanimement reconnu comme impropre au déploiement ; cependant, diminuer le volume d'informations en gérant les réservations au niveau d'équipements tiers, conduit à des solutions admissibles du point de vue de certains (EuQoS), mais insuffisantes pour d'autres (AQUILA, TEQUILA) ;
- de même, pour les contrats, la partie technique reste ouverte à plusieurs définitions, notamment quant aux performances, à la portée et à l'assurance du service fourni. L'absence de concept conduit certains à définir des cadres de solutions indépendants du choix (QBone, EuQoS) ; d'autres, tels que nous le montrerons ci-après, proposent une définition qui sert de fondement à leur solution (TEQUILA, MESCAL).

---

[86] http://qos.internet2.edu/wg/

[87] NGNI : *Next Generation Networks Initiative*. http://www.ngni.org/qos.htm

[88] QBone : *QoS Bone*. http://qbone.internet2.edu/

[89] Service analogue à celui d'une ligne louée.

[90] AQUILA : *Adaptive Resource Control for QoS Using an IP-based Layered Architecture*

[91] TEQUILA : *Traffic Engineering for Quality of Service in the Internet, at Large Scale*

[92] CADENUS : *Creation and deployment of end-user services in premium IP networks*

[93] MESCAL : *Management of End-to-end Quality of Service Across the Internet at Large*. http://www.mescal.org/

[94] EuQoS : *End to end QoS support over heterogeneous networks*. http://www.euqos.org/

Nous résumons maintenant les principales réponses apportées aux problèmes de provisionnement et de disponibilité des ressources.

3.2.2. Réponses au problème de provisionnement

L'enjeu du provisionnement d'un service consiste à ajuster le dimensionnement des ressources du réseau avec les besoins potentiels des clients.

Pour le multi domaines, le provisionnement revêt deux aspects :

- le premier, que nous n'avons pas abordé, est lié à la gestion des ressources internes (lien, routeur, etc.) de chaque domaine ;
- le deuxième aspect, que nous avons considéré, est lié à la découverte et au choix de concaténation des classes de services déployées sur le chemin de donnée.

3.2.2.1. Provisionnement intra domaine

Le sur-dimensionnement des ressources internes est une solution au provisionnement intra domaine. Il assure une garantie de service entre les points d'entrée et de sortie du domaine, mais induit une sous-utilisation des ressources, coûteuse pour le fournisseur.

Les projets AQUILA, TEQUILA et CADENUS ont abordé le problème différemment, suivant deux approches, orientée agent distribué et orientée politique :

- la gestion orientée agent consiste à configurer automatiquement des équipements à partir d'une base de données. Dans AQUILA, [Enge03] propose une gestion globale des ressources du domaine via un agent assurant la surveillance, le contrôle et la distribution des ressources dans le réseau ;
- la gestion orientée politique[95] offre une plus grande flexibilité [Pujo04] : elle repose sur un serveur de règles qui interprète et combine des informations enregistrées dans des bases de données circonstancielles pour prendre une décision quant à la politique à adopter ; les directives sont envoyées aux nœuds du réseau dans un second temps. L'intérêt est de permettre la mise en place en temps réel de services supportant par exemple de nouvelles normes de QdS ou de sécurité. Le protocole COPS[96] [Durh00] et son extension COPS-PR[97] [Chan01] ont été conçus pour assurer le déploiement de ces directives. CADENUS et TEQUILA suivent cette approche : les politiques de CADENUS sont déployées jusqu'au niveau des équipements réseaux [Corte03]. Celles de TEQUILA visent à l'ingénierie de trafic[98] [Trim01] [Trim02].

3.2.2.2. Provisionnement multi domaines

Dans le cas du multi domaines, la difficulté est d'assurer la satisfaction du service, lorsque la portée du SLS implique plusieurs domaines a priori hétérogènes quant à leur capacité à offrir de la QdS.

Ceci impose de définir comment découvrir et concaténer les classes de services disponibles sur le chemin qui sera emprunté par les données.

---

95 *Policy based Networking* (PBN)

96 COPS : *Common Open Policy Service protocol*

97 COPS-PR : COPS *Provisionning*.

98 Routage intra domaine orienté contraintes, emplacement de tunnels MPLS, …

Deux points de vue ont été adoptés pour y répondre :

- le premier aborde la gestion du multi domaines d'un point de vue *fournisseur* : c'est ce dernier qui effectue la découverte et le choix de concaténation des classes de services sur le chemin de donnée, en amont de la requête de service. Son objectif est d'établir un compromis visant d'une part, à optimiser la capacité globale du domaine et de ses liens externes, et d'autre part à maximiser le nombre de clients potentiels ;
- le second point de vue, que nous avons adopté, part du constat que le concept de gestion du multi domaines n'est pas mature ; suivant ce point de vue, il incombe à l'utilisateur de « construire » dynamiquement le service de bout en bout répondant aux besoins applicatifs, en coopération avec les gestionnaires des domaines traversés. L'objectif est de minimiser l'utilisation des classes de services les plus coûteuses.

Point de vue fournisseur

Suivant ce point de vue, le fournisseur cherche à offrir des services de bout en bout aux caractéristiques contractuelles (performances, assurance, portée) pré-définies, allant potentiellement au delà de son domaine.

Les réponses au choix de concaténation sont multiples car il n'y a pas (encore aujourd'hui) de consensus sur les concepts de *performances*, d'*assurance* et de *portée du service* :

- les performances peuvent concerner différents aspects tels que :
  - la QdS associée au transfert des données, exprimable en termes qualitatifs, dans EuQoS par exemple, ou en termes quantitatifs, dans MESCAL ou TEQUILA ;
  - le succès de l'invocation du service ;
- l'assurance de réalisation de ces performances peut se traduire par une garantie ferme, une effectivité en moyenne, etc. ;
- enfin, la notion de portée du service n'est pas toujours précisée, et le client peut espérer obtenir les performances pour toutes les destinations.

Les principaux types de propositions sont les suivants.

[Mant04] propose une solution basée sur l'établissement préalable de tunnels de bout en bout (*pipe*[99]) pour lesquels sont effectuées des réservations de bande passante. Ces *pipes* ne concernent que les domaines fournisseurs, pour une gamme de services pré-définis et identiques pour tous. Le choix de concaténation se réduit au choix d'un *pipe* compatible avec la requête. Cette solution a pour inconvénient majeur de supposer une forte homogénéité des services fournis ; l'automatisation de l'établissement des *pipes* reste un problème ouvert, nécessaire pour rendre l'approche dynamique.

Le projet MESCAL définit une phase de souscription de contrat pour négocier les caractéristiques techniques des différentes classes de service. Celles-ci s'appuient sur la proposition initiale de [Gode01] qui décrit les performances de QdS en termes de délai, de taux de pertes, …, et indique aussi la portée du service[100]. MESCAL opte pour une garantie faible, dont la sémantique n'est pas définie. Il adopte également ce modèle de contrat pour les accords de *peering* [Howa05].

---

[99] Agrégat de trafic d'une même classe de service qui va d'un routeur de sortie d'un domaine à un routeur d'entrée d'un autre domaine (non forcément adjacent).

[100] Typiquement un préfixe IP identifiant un ou plusieurs réseaux.

Sa solution au problème de concaténation repose sur le concept de classe de QdS étendue (*e-QC*), défini récursivement comme la concaténation d'une classe locale (*l-QC*) et d'une e-QC[101] d'un domaine adjacent. Face aux demandes de souscriptions, le choix de concaténation s'effectue en deux étapes : toutes les concaténations de l-QC avec les e-QC des domaines adjacents sont d'abord évaluées ; ensuite, l'une de celles qui satisfont le contrat est retenue.

Dans le projet EuQoS, les notions de portée et d'assurance ne sont pas explicitées et les performances de QdS sont exprimées de façon qualificative[102]. La solution au problème de concaténation repose sur la connaissance préalable de chemins de bout en bout déployant une même classe de service sur chaque tronçon (*end to end QoS path*). A chaque chemin est associée une bande passante maximale non réservée. Le problème se réduit alors au choix d'un chemin adéquat lors du contrôle d'admission effectué suite à l'invocation du service.

Notons que les deux dernières solutions s'appuient sur l'hypothèse d'un routage à QdS étendu au multi domaines, ce qui nécessite une extension du protocole de routage BGP[103] ; [Cris02] [Cris03] posent les bases des modifications nécessaires.

Point de vue utilisateur

Les propositions basées sur le point de vue utilisateur font l'hypothèse de contrats entre clients et fournisseurs, dont les performances et la portée ne sont pas définies à l'avance : c'est à l'utilisateur de découvrir les services disponibles sur le chemin, de les choisir, puis de les invoquer. Plusieurs propositions ont été faites en ce sens.

[Fuze03] propose des mécanismes de découverte des paramètres de QdS pour une classe de service dans le but de fournir, avant la souscription, des informations sur la QdS disponible. Il considère quelques classes de service standard.

Dans nos travaux [Auri04b], une requête de QdS exprime les besoins de l'utilisateur en termes de paramètres de QdS sans préciser la classe de service requise. La découverte des classes de service et de leurs performances disponibles sur le chemin choisi par BGP, est alors effectuée par le biais d'une signalisation adéquate. La *meilleure* concaténation de classes qui, couplée au Transport, permet de satisfaire la requête, est ensuite retenue, et la réservation est initiée. L'objectif est de minimiser l'utilisation des ressources les plus coûteuses en bande passante. Cette solution est indépendante du type de version de BGP.

[Yang05] poursuit le même objectif. Sa proposition est basée sur l'utilisation de *vecteurs de services* conduisant à la possibilité de choisir différents PHB successifs, afin d'obtenir une concaténation répondant aux besoins ciblés. A la différence de nous, il n'aborde pas explicitement certains problèmes liés au multi domaines, ni la prise en compte du Transport dans l'évaluation des performances de bout en bout.

3.2.3. Réponses au problème de disponibilité des ressources

L'assurance de disposer des ressources sur le chemin de donnée nécessite la mise en œuvre d'un contrôle d'admission pour les requêtes de connexion (CAC).

---

[101] Eventuellement réduite à une l-QC.

[102] RT : *Real Time*, et NRT : *Non Real Time*

[103] BGP : *Border Router Protocol*

On distingue deux aspects dans ce contrôle :

- le contrôle de la disponibilité des ressources à l'intérieur de chacun domaine ; ce contrôle, que l'on appelle le CAC intra domaine, est de la responsabilité de chaque domaine ;
- le contrôle de la disponibilité pour chacun des liens reliant les différents domaines, au regard des accords de *peering* et de l'état courant d'utilisation ou de réservation.

Pour le multi domaines, le CAC couple ainsi les CAC intra domaine de chacun des domaines traversés, et ceux effectués sur chacun des liens inter-domaines.

Afin d'assurer une garantie satisfaisante, le contrôle d'admission doit être complété par un mécanisme de protection du trafic conforme.

3.2.3.1. Protection du trafic conforme

Les solutions proposées, dont la notre, reposent sur une classification, une réservation (maintien d'état) et un *policing* par flux en entrée du réseau, au niveau du routeur de bordure. Par contre, les routeurs de cœur ne considèrent que des classes.

Pour le multi domaines, la solution peut aussi impliquer les routeurs d'entrée des domaines intermédiaires ; en ce cas, le bénéfice sur la garantie se fait au détriment de la résistance au facteur d'échelle de l'approche.

3.2.3.2. CAC intra domaine

Plusieurs travaux ont traité le besoin d'un CAC intra domaine pour gérer les ressources au sein d'un domaine DiffServ.

En cœur de réseau, deux approches sont envisageables :

- la première est basée sur une réservation de ressource par classe en tout ou partie des routeurs de cœur impliqués dans le chemin de donnée [West02] [Kara04] [Bade05]. Elle implique donc le maintien d'un état par classe en cœur de réseau. L'acceptation d'une requête se fait en fonction des ressources disponibles, compte tenu des réservations déjà effectuées ; un protocole de signalisation est nécessaire à la propagation de la requête auprès des routeurs concernés ;
- la deuxième approche, orientée mesures, est basée sur une estimation périodique de la disponibilité des ressources et d'une prédiction de la charge qu'induirait l'acceptation d'une nouvelle requête. Suivant le niveau de charge de l'élément considéré, une requête est acceptée si son profil de trafic ne conduit pas à un risque important de saturation ; avec cette approche, les routeurs de cœur n'ont pas à maintenir d'état de réservation.

Dans TEQUILA, [Myko03] illustre cette deuxième approche par l'application de trois politiques, distinguées selon la charge estimée sur la route. La première, attachée à une situation nominale, conduit à accepter les requêtes dont le profil n'engendrerait pas de congestion. La seconde, consécutive à la survenue d'une congestion, rejette toute nouvelle demande. La troisième politique correspond à une phase de relaxation précédant le retour à la situation nominale, durant lequel le contrôle d'admission est plus restrictif.

Il laisse ouverte la façon de prédire la charge induite par l'acceptation d'une requête et la description du profil de trafic.

Deux types de propositions basées sur le principe de l'essai/échec fournissent une réponse :

- dans la première [Fred01] [Hass02], le flux de données est émis immédiatement. Des mesures effectuées sur ce flux conduisent à conclure sur son admissibilité ; cette approche n'implique pas la description du profil de trafic par l'utilisateur ;
- dans la deuxième [Bres00] [Gane05], un flux test semblable au profil annoncé est émis avant la réponse au contrôle d'admission. Comme précédemment, le CAC résulte de la mesure de la QdS expérimentée par le flux.

3.2.3.3. CAC inter domaines

Pour l'inter domaines, la difficulté est de prolonger le contrôle d'admission de domaine en domaine, en tenant compte des liens inter domaines. Deux approches sont explorées :

- la première prolonge la vue IntServ en impliquant seulement les routeurs de bordure vis à vis du maintien d'état par flux ; ceux-ci sont successivement consultés pour assurer le CAC au sein de leur domaine et sur le lien inter domaines ; la réservation qui s'en suit engendre le maintien d'un état par flux ou par agrégat. Les routeurs de cœur ne maintiennent au plus[104] qu'un état par agrégat. Cette approche a en particulier été étudiée dans le projet AQUILA [Sals03] et elle constitue actuellement le modèle de QdS pour le multi domaines pris en considération par NSIS[105] [Bade05] ;
- la deuxième approche, que nous avons suivie, se démarque plus nettement d'IntServ, en considérant les domaines comme des entités gérées de manière fortement hétérogène. La gestion de chaque domaine est abordée de manière plus hiérarchique par le concept de *Bandwidth Broker* (BB). En particulier, le contrôle d'admission intra domaine n'est plus de la responsabilité directe des routeurs, mais des BB. Pour le multi domaines, une coopération entre les BB est nécessaire pour assurer le CAC de bout en bout (Figure 6).

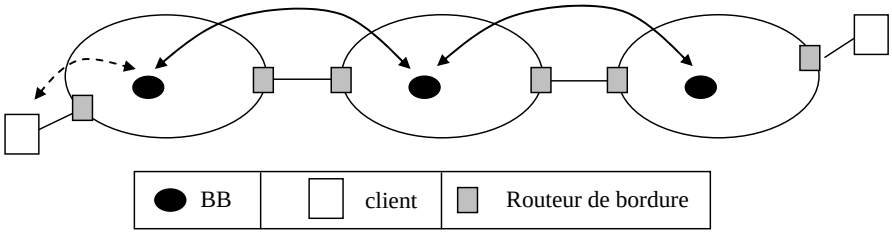


Figure 6 : Modèle à base de *Bandwidth Broker* (BB)

[Nich99] est la première solution proposée suivant cette approche. Son objectif est d'obtenir le même type de garanties qu'IntServ mais sans impliquer les routeurs de cœur dans le contrôle d'admission et la réservation. Chaque réservation acceptée engendre le maintien d'états au niveau des routeurs de bordure, qui doivent être périodiquement rafraîchis.

Il introduit plusieurs scénarios, basés sur des accords de *peering* statiques ou négociables par requête. Dans le cas statique, une requête n'est acceptée par un BB que si ses ressources locales et les réservations effectuées sur le lien de sortie le permettent. Dans le cas dynamique, la quantité de trafic de chaque classe n'est pas spécifiée dans l'accord de *peering*, mais négociée lors de la requête. Une telle dynamicité est exclue dans l'Internet actuel.

---

[104] Dans le cas du CAC intra domaine basé réservation.

[105] NSIS, que nous présenterons en section III, contribue essentiellement à la signalisation pour l'Internet ; par ce biais, il contribue indirectement à l'élaboration des modèles de QoS inter domaines.

Plusieurs travaux proposent d'améliorer [Nich99] vis à vis de ses limites en termes de passage à d'échelle (liées au nombre et au maintien des états de réservation), et de risque de sous utilisation des ressources (dans le cas statique).

- Dans le projet QBone, [Terz99a] s'affranchit de la nécessité des rafraîchissements en ajoutant aux requêtes les dates de début et de fin de la réservation pour le flux considéré ; ceci permet de ne conserver qu'un état global de réservation par classe et par lien inter domaines. Face au risque de sous utilisation des ressources inter domaines, il propose une sur allocation dont il contrôle les conséquences potentielles par monitorage au niveau des routeurs de bordure ; en cas d'alerte, on tente de réajuster les accords de *shaping/policing* entre routeurs de bordure du lien inter domaines considéré ; en cas d'échec, une alerte est émise en direction des sources [Terz99b]. Cela pré-suppose donc la renégociation dynamique des accords de *peering*, ce qui dans les faits est difficile.

- [Mant04] répond au problème du passage à l'échelle, en basant sa solution sur l'établissement préalable de tunnels de bout en bout (*pipe* précédemment décrits pour le provisionnement) ; les requêtes de service sont soumises à un contrôle d'admission appliqué aux extrémités[106] du *pipe*. En cas de succès, ce contrôle n'est pas propagé aux domaines suivants, du fait de la réservation préalable. En cas d'échec, une balance de charge sur un autre *pipe* est tout d'abord envisagée ; si elle n'aboutit pas, une nouvelle réservation est négociée avec le BB du domaine suivant via une signalisation appropriée. L'inconvénient de cette solution résulte du fort couplage exigé entre les domaines, ce qui en limite la perspective de déploiement.

D'autres travaux prolongent [Nich99] par la proposition d'architectures plus complètes.

- EuQoS propose une solution globale à base de BB (RM[107] dans la terminologie EuQoS) pour lesquels il définit une architecture qui identifie trois blocs fonctionnels : le provisionnement, le CAC et la signalisation ; il distingue deux niveaux de fonctionnalités, dépendant (NTD[108]) ou non (NTI[109]) des technologies sous jacentes.

Suite à une requête, le contrôle d'admission de bout en bout se déroule en deux phases :

1) au niveau NTI, choix d'un *end to end QoS path*[110] satisfaisant la bande passante requise pour la classe de service souhaitée ;
2) si un chemin existe, invocation récursive (de RM en RM) du contrôle d'admission sur le lien inter domaines et du CAC intra domaine au niveau NTI, relayée au niveau NTD par une fonction d'allocation de ressources spécifique à la technologie sous jacente.

- Nos travaux complètent [Nich99] par la découverte des classes de service en même temps que le CAC [Auri04b]. Contrairement à EuQoS, une requête de QdS n'identifie pas la classe de service requise mais une bande passante et des paramètres de performances souhaités.

Le but est d'obtenir un chemin présentant une classe de service homogène, qui couplée au Transport, satisfait les paramètres de QdS requis.

---

106 C'est à dire dans le domaine et sur le lien de sortie.

107 RM :*Resource Manager*

108 NTD : *Network Technology Dependant*

109 NTI : *Network Technology Independent*

110 Défini précédemment comme un chemin de bout en bout déployant une même classe de service sur chaque tronçon.

Une requête de bande passante pour des paramètres de performance, initie le traitement qui se déroule en trois phases :

1) détermination des performances de chacune des classes du chemin et pré-réservation des ressources disponibles ;
2) choix du meilleur couple **(**Transport, classe de service IP**)** compatible avec les performances requises ;
3) confirmation de la réservation pour la classe de service choisie et libération des pré-réservations non retenues.

3.3. Conclusions

Plusieurs réponses ont été avancées aux deux problèmes majeurs que sont le provisionnement et la disponibilité des ressources, pour l'obtention d'une QdS IP dans l'Internet multi domaines.

Aucune n'a pour l'instant conduit à une solution suffisamment satisfaisante pour être déployée par les opérateurs à l'échelle de l'Internet. Cependant, chacune fournit un éclairage supplémentaire sur la compréhension globale du problème.

En conclusion de cet état de l'art, et pour chacun des problèmes, il ressort les points suivants.

- Vis à vis du provisionnement :

– le problème a été abordé suivant deux points de vue : celui du fournisseur offrant des services aux performances pré-définies, et celui de l'utilisateur cherchant à acquérir la connaissance de ces performances avant de sélectionner le service. Nous avons adopté le second point de vue dans l'optique de répondre, au plus près des capacités du réseau, aux besoins des applications à QdS, en tenant compte de l'impact du Transport ;
– les solutions proposées impliquent un couplage plus ou moins fort entre les différents domaines. Nos travaux se sont placés dans l'hypothèse d'un couplage faible, qui laisse davantage de latitude vis à vis de l'hétérogénéité d'administration des domaines ;
– ces solutions s'appuient ou non sur l'hypothèse d'un routage inter domaines à QdS ; notre proposition est indépendante de cette fonctionnalité ;

- Vis à vis de la disponibilité des ressources :

– le CAC à l'échelle du multi domaines doit s'appuyer sur l'hypothèse d'une gestion indépendante des CAC intra domaine ; nos solutions adoptent ce point de vue ;
– sur chaque domaine, le CAC peut être mis en œuvre au sein de BB, ou bien de manière distribuée sur des routeurs du chemin de donnée ; nous avons opté pour le premier modèle, qui semble la solution autorisant le plus de flexibilité vis à vis de l'indépendance dans la gestion des domaines.

La motivation commune à tous nos choix est de prendre en compte le modèle le plus faiblement contraint pour la gestion de l'Internet multi domaines, en particulier vis à vis du routage ; elle peut cependant tirer partie des avantages procurés par des modèles à base de routage inter domaines à QdS.

A l'image de la majorité des travaux, nos propositions sont à étendre pour prendre en compte le fait que l'Internet du futur sera surement constitué de domaines hétérogènes quant à leur capacité à garantir de la QdS. Cette ambition fait partie de notre prospective.

III. La signalisation pour la QdS dans l'Internet multi domaines

1. Introduction

Plusieurs modèles de gestion de la QdS ont été proposés pour l'Internet multi domaines. Ces modèles diffèrent par l'angle d'attaque du problème[111], et leurs propositions de solutions sont différentes ; cependant, la plupart des solutions proposées nécessitent une signalisation.

C'est en ce sens que nous avons adressé le thème de la signalisation, pour mettre en œuvre, suivant un point de vue utilisateur, nos propositions de gestion de la QdS dans un modèle d'Internet multi domaines, doté de BB au sein de chaque domaine.

Le besoin en signalisation ne concerne pas uniquement le thème de la QdS dans l'Internet, mais également la mobilité, la sécurité et la métrologie. Ainsi, face à ces multiples besoins, un groupe de travail de l'IETF, NSIS[112], s'est récemment constitué dans le but de définir un cadre générique pour la conception de protocoles de signalisation [Hanc05a] [Fu05].

Pour aboutir à une solution modulaire, ce cadre distingue deux niveaux, NTLP[113] et NSLP[114] :

- le niveau NTLP est dédié au transport de la signalisation entre entités de NSIS, indépendamment de la nature de cette signalisation (orientée QdS, sécurité, …) ;
- le niveau NSLP est dédié aux applications de signalisation, spécifique de l'intérêt ciblé ; chaque application définit le format et l'enchainement des messages échangés.

NSIS définit deux étapes dans sa logique de travail :

- dans la première (en cours), il fait l'hypothèse, comme pour RSVP, que les entités de signalisation sont toutes localisées sur des équipements impliqués dans le chemin de donnée ; cette signalisation est dite *couplée au chemin de donnée ;*
- dans la deuxième étape[115], NSIS étudie le besoin en signalisation *découplée du chemin de donnée*. Ceci répond aux cas où les entités de signalisation sont déployées sur des équipements non impliqués dans le chemin des données, comme peuvent l'être les BB.

La première cible de NSIS est la signalisation pour la QdS. NSIS vise à définir un cadre indépendant du modèle de QdS retenu, et se fixe l'objectif à moyen terme d'une solution valable pour les deux approches, couplée et découplée du chemin de donnée [Hanc05b].

Au niveau NTLP, le protocole GIST[116] (anciennement appelé GIMPS) est actuellement en cours d'étude [Schu05] [Tsen05] pour le transport de signalisation couplée au chemin de donnée. GIST assure les échanges de proche en proche et dans les deux sens du flux de données. Il est découplé des applications de NSLP, qui accèdent au service via une API ; il utilise les protocoles de Transport et de sécurité existants.

Sur les bases du protocole RSVP, la signalisation pour la QdS constitue actuellement le premier cas d'étude du niveau NSLP.

---

111 Point de vue utilisateur ou fournisseur par exemple.

112 NSIS : *Next Step In Signaling Protocol*

113 NTLP : *NSIS Transport Layer Protocol*

114 NSLP : *NSIS Signaling Layer Protocol*

115 Initiée tout récemment [Hanc05b]

116 GIST : *General Internet Signaling Transport protocol* (anciennement GIMPS : *General Internet Messaging Protocol for Signaling)*

Nos contributions à la signalisation n'ont pas été menées dans le cadre NSIS ; cependant, elles incluent des fonctionnalités typiques du niveau NSLP ; leur généralité vis à vis de la localisation des BB, permet d'envisager leur prise en compte au niveau NTLP pour une signalisation découplée du chemin de donnée ; cela fait partie de notre prospective.

La suite de cette section est structurée comme suit. Les problèmes associés à la signalisation sont décrits dans le paragraphe (2) suivant. Les propositions de solutions sont exposées au paragraphe (3).

2. Problèmes

Pour l'Internet multi domaines, les solutions aux problèmes de provisionnement et de CAC nécessitent une signalisation pour :

- la découverte des classes de service et de leurs performances sur le chemin de donnée,
- l'évaluation de la disponibilité des ressources sur ce chemin, et leur réservation éventuelle.

2.1. Vis à vis du provisionnement

La découverte des classes de service et de leurs performances peut être envisagée de deux façons (cf. paragraphe 3.2.2.2 de la section II précédente) :

- soit par acquisition dynamique au moment de la requête de QdS, pour que l'information soit disponible en émission et confrontée aux besoins exprimés dans la requête ; cela nécessite le parcours d'un chemin assurant la connectivité inter domaines, pour obtenir de chacun de ceux-ci les caractéristiques des classes de service qu'il supporte ;
- soit indépendamment de la requête de QdS, par accès dans la table BGP à des informations sur les capacités en QdS des différents chemins [Cris02] [Cris03] [Xiao02] [Xiao04] ; c'est au moment du contrôle d'admission que le choix d'un chemin compatible avec la demande de QdS est effectué[117]. En terme de signalisation, le problème consiste à définir comment étendre le protocole BGP.

Nous avons contribué à la signalisation requise pour le provisionnement suivant la première approche. Notre proposition, indépendante de la version de BGP, couple dans une même signalisation les informations relatives à la découverte des services et au contrôle d'admission sur le multi domaines.

Deux principaux arguments nous ont conduits à ne pas adopter l'approche basée sur BGP :

- compte tenu de la sensibilité de BGP, sa modification en vue de propager des informations sur la QdS constitue un risque pour la stabilité de l'Internet ;
- en outre, les propositions d'extension de BGP pour la QdS visaient initialement à satisfaire deux préoccupations divergentes quant à leur échelle de temps :
  - d'une part, la connaissance de la capacité en bande passante des chemins vis à vis des classes de service, selon une approche quasi statique ;
  - d'autre part, une évaluation actualisée de la disponibilité de la bande passante sur les chemins à classe de service, selon une approche dynamique.

[117] L'obtention de QdS s'appuie donc sur un routage à QdS.

Par ailleurs, aucune n'abordait l'acquisition d'informations sur les performances effectives des chemins à QdS.

Notre souci étant de déterminer les performances associées au(x) chemin(s) à QdS disponible(s) afin d'effectuer le choix optimum du couple (Transport, classe de service IP), il était nécessaire de définir une signalisation permettant d'accéder à cette information.

Notons cependant que des travaux en cours, en particulier dans le cadre du projet EuQoS, poursuivent l'approche basée sur BGP pour compléter l'information sur la capacité des chemins à QdS par des données sur les performances.

2.2. Vis à vis de la disponibilité des ressources

Pour le multi domaines, l'évaluation de la disponibilité des ressources porte sur chacun des domaines et sur chacun des liens inter domaines du chemin de donnée.

En intra domaine, le CAC peut être effectué selon une approche orientée réservation ou estimation. Pour le multi domaines, deux approches sont envisagées :

- dans le premier cas, le CAC implique directement les routeurs de bordure successifs du chemin de donnée, il s'appuie donc naturellement sur une signalisation inter domaines couplée à ce dernier (telle que le cible NSIS dans sa première phase) ;
- dans le deuxième cas, le CAC implique la coopération des BB des domaines empruntés ; ces BB étant extérieurs au chemin de donnée, la signalisation requise est naturellement découplée du chemin de donnée. En outre, pour satisfaire les réservations, les BB doivent pouvoir configurer certains routeurs du tronçon intra domaine dont ils ont la charge[118] : une signalisation intra domaine est ici nécessaire.

Notre proposition de signalisation pour le provisionnement répond également au deuxième cas présenté ci-dessus.

2.3. Vis à vis des modèles à base de BB

Les solutions à base de BB posent deux problèmes principaux :

- un BB est supposé avoir une « vue » de son domaine qui lui permet de statuer sur la disponibilité des ressources entre tout couple de routeurs de bordure de son domaine. Pour une requête particulière, le problème est alors de déterminer les deux routeurs de bordure par lesquels passe le chemin de donnée ;
- pour la propagation de la requête au BB suivant, le problème est d'identifier ce dernier.

Notre proposition de signalisation traite explicitement ces deux problèmes, en supposant, dans un premier temps, l'unicité des BB dans chaque domaine.

La Figure 7 permet d'illustrer ces deux problèmes, par exemple pour le BB_2.

Recevant une requête du BB_1, le BB_2 doit pouvoir identifier les routeurs Rb_{21} et Rb_{22} pour répondre au contrôle d'admission dans le domaine 2 et sur lien avec le domaine adjacent. La propagation de la requête au BB_3 impose que celui soit identifiable (par exemple) par BB_2.

[118] Typiquement les routeurs de bordure.

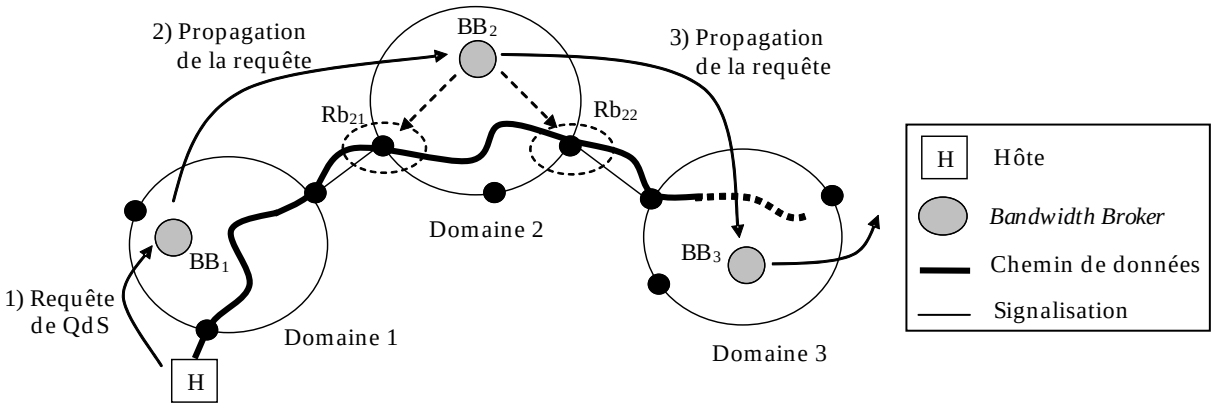


Figure 7 : Problème associé au modèle à base de BB

3. Propositions

La signalisation pour la QdS dans l'Internet multi domaines a logiquement été abordée en fonction du modèle de QdS retenu.

Ces modèles conduisent à deux types de signalisation : couplée au chemin de donnée (SCCD[119]), ou découplée de ce chemin (SDCD[120]). C'est sous cet angle que nous présentons l'état de l'art sur la signalisation.

Les propositions décrites ci-après (dont la notre) adressent toutes le problème de la signalisation pour le CAC multi domaines, sans faire d'hypothèse sur le type de CAC intra domaine. Mis à part la notre, elles s'appuient sur l'hypothèse d'un provisionnement effectué en amont, notamment pour certains par une extension du protocole BGP.

3.1. L'approche SCCD (signalisation couplée au chemin de donnée)

3.1.1. Introduction

Dans l'approche SCCD, les entités de signalisation sont obligatoirement sur des équipements du chemin de donnée. Cependant, entre deux entités, les données de signalisation peuvent suivre un autre chemin que celui empruntée par les données.

L'inconvénient majeur de l'approche SCCD est que, par principe, le CAC est traité de façon distribuée sans aboutir à une vue globale au niveau de chaque domaine. La décision finale vis à vis d'une requête résulte de l'unanimité des décisions locales, et de l'homogénéité du comportement des routeurs concernés.

Ses avantages sont de deux types :

- par principe, la route pour assurer la signalisation est connue (c'est celle des données), et tout changement de route est naturellement détecté ;
- l'interaction avec les fonctions de contrôle du trafic (*policing*, *scheduling*, …) se faisant localement, il n'y a pas besoin de signalisation pour la configuration des routeurs ; comparativement aux solutions nécessitant une telle signalisation (celles de l'approche SDCD), le temps de réponse est mécaniquement plus court.

---

[119] *Path coupled signaling* dans la désignation NSIS

[120] *Path decoupled signaling* dans la désignation NSIS

3.1.2. Propositions

3.1.2.1. De RSVP à NSLP-QoS

L'illustration typique de l'approche SCCD est le protocole de signalisation RSVP. RSVP a été initialement défini dans le contexte IntServ pour supporter un contrôle d'admission basé sur la réservation, au niveau de chacun des routeurs du chemin de donnée.

La principale cause d'inadéquation d'RSVP au contexte du multi domaines tient à son mode de réservation, nécessairement de bout en bout : il est obligatoirement activé par les hôtes d'extrémité et il interdit une réservation limitée à un tronçon du chemin de donnée. En outre, il impose que ces réservations soient effectuées par les récepteurs uniquement.

Actuellement en cours d'étude, la proposition NSLP-QoS vise dépasser ces limites [Mann05] [Ash05]. Aujourd'hui, NSLP-QoS considère plusieurs scénarios de mise en œuvre d'une réservation (par flux), à l'initiative de l'émetteur, du récepteur, ou bidirectionnelle. Il considère également la possibilité d'agrégation des réservations. Il étudie aussi le cas des changements de route et le re routage de la signalisation en conséquence.

3.1.2.2. BGRP

BGRP[121] est un protocole de signalisation conçu pour un environnement multi domaines à base de réservation sur les liens inter domaines [Pan00]. Son objectif est la résistance au facteur d'échelle vis à vis du nombre des messages échangés et des états de réservation maintenus sur les équipements concernés, qui sont les routeurs de bordure des domaines.

BGRP agrège, au niveau des routeurs de bordure, les réservations portant sur des classes de trafic en fonction de régions de destinations. Comparativement à RSVP, dont les solutions d'agrégation portent sur des flux et concernent tous les routeurs, BGRP diminue le nombre des états de réservation ainsi que la charge des messages de rafraîchissement de ces états.

BGRP a été étudié dans l'architecture proposée dans le projet AQUILA. Les gains qu'il procure vis à vis du problème de la mise à l'échelle sont évalués dans [Pan00] [Sals03].

3.2. L'approche SDCD (signalisation découplée du chemin de donnée)

3.2.1. Introduction

Dans l'approche SDCD, les entités de signalisation sont localisées indépendamment des équipements du chemin de donnée.

L'avantage (et la motivation) de l'approche est qu'elle permet un traitement du CAC en ayant une vue globale au niveau de chaque domaine. La décision finale vis à vis d'une requête ne dépend que de l'unanimité des BB, et non des routeurs de bordure du chemin.

En contrepartie, elle doit répondre aux problèmes déjà énoncés qui sont :

- d'une part, que les BB soient identifiables,
- d'autre part, que chaque BB connaisse les points d'entrées et de sortie du tronçon du chemin des données traversant son domaine, et sa modification éventuelle.

[121] BGRP : *Border Gateway Reservation Protocol*

En outre, elle nécessite une signalisation complémentaire pour configurer les routeurs de bordure dans le but de protéger les flux conformes.

3.2.2. Propositions

3.2.2.1. SIBBS

SIBBS[122] est le protocole de signalisation défini par le QBone pour la mise en œuvre du CAC dans un environnement multi domaines à base de BB [QSDT02].

SIBBS est basé sur l'échange de deux PDU[123] principaux : RAR et RAA[124]. Le PDU RAR contient les informations relatives à la requête de QdS (classe de service et bande passante requises notamment) ; il peut être engendré par l'hôte initiateur de la requête, par un BB ou par un élément tiers faisant office de proxy (applicatif ou IP). Le PDU RAA contient la réponse à un RAR. La communication des RAR et RAA entre deux BB est supposée fiable[125].

SIBBS parcourt chacun des BB du chemin inter domaines, pour permettre à celui ayant reçu la requête initiale de statuer sur la disponibilité de ressources tout au long du chemin.

Les principes de base du protocole sont les suivants :

- recevant un RAR, un BB évalue la disponibilité de ses ressources sur le tronçon du chemin le traversant et sur le lien de sortie vers le domaine suivant. Si elles sont compatibles avec la requête, il réserve les ressources demandées auprès du routeur de bordure par lequel les données abordent son domaine, puis émet un RAR à destination du BB suivant ;
- en supposant que les ressources sont disponibles tout au long du chemin, le processus se poursuit de façon récursive jusqu'au BB de l'hôte destinataire ;
- ce dernier retourne alors un RAA positif au précédent BB, et ainsi de suite jusqu'au BB initial, qui conclut positivement sur l'admission de la requête.
- enfin, le maintien des réservations est assuré via l'émission périodique de messages de rafraîchissement.

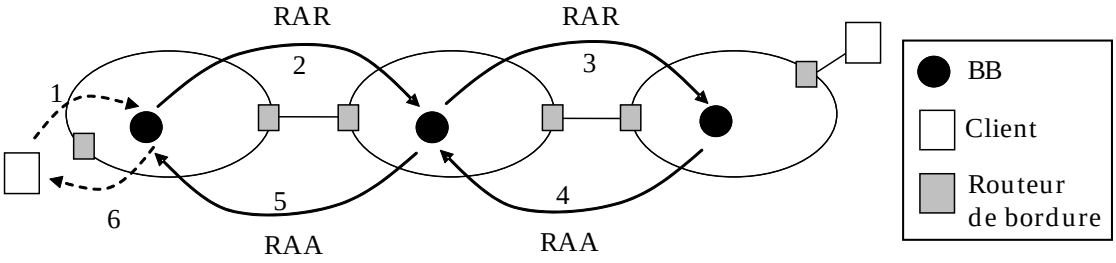


Figure 8 : Modèle du protocole SIBBS

La communication entre BB et routeur de bordure (pour la configuration de ce dernier) n'est pas précisé dans SIBBS. Elle peut s'appuyer sur la proposition COPS-PR rappelée ci-après.

SIBBS ne spécifie pas comment il répond au problème d'identification du BB suivant. Il suppose que les BB ont accès à la table BGP, ce qui leur permet d'acquérir la connaissance des routeurs d'entrée et de sortie de leur domaine.

---

[122] SIBBS : *Simple Inter domain Bandwidth Broker Signalling protocol*.

[123] PDU : *Protocol Data Unit*

[124] RAR : *Ressource Allocation Request* / RAA : *Ressource Allocation Answer*

[125] En pratique, elle est assurée par TCP.

3.2.2.2. COPS-SLS

COPS-SLS [Tran02] [Tran03] est une proposition d'adaptation du protocole COPS pour la gestion de SLS dans un environnement multi domaines.

Principes de COPS

COPS [Duhr00] a été défini pour permettre la gestion de réseau à base de politiques [Pujo04]. Il est basé sur les concepts de PDP et PEP[126] :

- un PDP est une entité logique qui produit des décisions politiques (sous formes de règles) pour lui même ou pour d'autres éléments du réseau ; une décision est suivie d'actions quand les conditions des règles sont vraies ;
- un PEP est le point d'application de ces règles (un routeur par exemple).

COPS est un protocole de type requête / réponse qui permet à un routeur (PEP) d'interroger son PDP pour acquérir les règles à appliquer suite à un événement tel que la réception d'un paquet de signalisation. COPS-PR [Chan01] est une extension de COPS visant à ce qu'un PDP puisse imposer l'application de règles à un PEP, sans requête au préalable.

COPS-SLS

COPS-SLS reprend les mêmes principes de récursivité dans les échanges inter BB que le protocole SIBBS. Chaque BB se conduit tour à tour comme un PDP vis à vis de l'émetteur de la requête qu'il reçoit, puis comme un PEP vis à vis du BB suivant.

Comparativement à SIBBS, COPS-SLS ajoute un certain nombre de fonctionnalités, comme la renégociation de classe de service en cas d'échec du CAC.

La communication entre BB et routeur de bordure est effectuée via COPS-PR.

COPS-SLS ne spécifie pas comment est identifié le BB suivant, ni comment sont déterminés les routeurs d'entrée et de sortie des domaines.

3.2.2.3. Autres propositions

EuQoS propose deux solutions (actuellement en cours d'étude) pour la signalisation sous jacente au CAC multi domaines :

- la première solution, inspirée de [Hanc05b], propose une adaptation pour détourner la signalisation NSIS vers les BB. En entrée d'un domaine, tout routeur de bordure (NSIS) intercepte les paquets de signalisation et les transmet à son BB (qu'il est supposé connaitre). Après traitement, le BB les retourne au routeur, qui les transmet au routeur de sortie du domaine (Figure 9). Le dialogue entre les BB s'effectue donc sans nécessiter de leur part la connaissance explicite de leur adresse respective ;

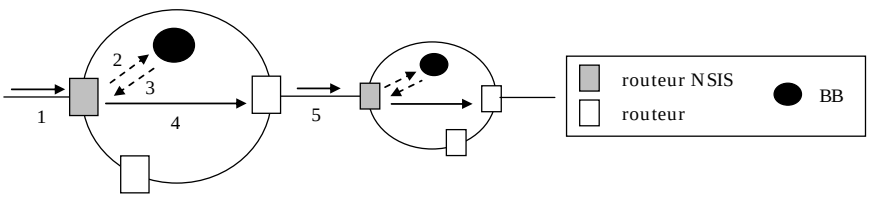


Figure 9 : Esquisse de la solution NSIS-RM-SNN pour la signalisation inter BB

[126] PEP: *Policy Enforcement Point* / PDP : *Policy Decision Point*

– la deuxième solution, à laquelle nous contribuons, vise à généraliser la proposition initiée dans nos précédents travaux pour l'identification du prochain BB et la détermination des routeurs d'entrée et de sortie [Auri04b]. Elle s'appuie sur les principes de base du protocole SIBBS, et ne poursuit pas l'approche originale pour la découverte des services, supposée effectuée, comme pour la première solution, via un BGP orientée QdS.

[Auri04b] propose une signalisation assurant conjointement la découverte des services et de leurs performances et la vérification de leur disponibilité. Il fournit également une première réponse aux problèmes laissés ouverts pour SDCD : l'identification du prochain BB et la détermination des routeurs d'entrée et de sortie. Cette réponse suppose la connaissance de l'adresse du BB pour les domaines adjacents et l'accès à la table BGP. Le protocole proposé est décrit dans l'Etape 3 de cette partie du mémoire.

4. Conclusion

Les principales propositions de protocole de signalisation pour la mise en œuvre du contrôle d'admission dans l'Internet multi domaines ont été présentées.

Deux types de signalisation sont actuellement en cours d'étude, en particulier au sein de l'IETF dans le cadre des travaux du groupe NSIS, SCCD et SDCD :

– la première est dans son principe contradictoire avec une gestion du CAC à base de BB[127], mais apporte une réponse naturelle au besoin d'une signalisation portant au bout du compte sur des équipements impliqués dans le chemin de donnée ;
– la deuxième découle des contraintes des modèles à base de BB ; c'est logiquement celle ci que nous avons ciblée dans nos travaux.

Comparativement aux autres signalisations qui suivent cette deuxième approche, notre proposition répond non seulement au besoin du CAC multi domaines et mais également à celui du provisionnement des services, vis de la découverte des classes de services et de leur performances sur le chemin de donnée.

A la différence de SCCD, SDCD pose le problème de l'identification des BB pairs et de la détermination des routeurs d'entrée et de sortie de chaque domaine, pour le chemin emprunté par les données. Nous avons proposé une solution à ce problème sous l'hypothèse de l'unicité des BB au niveau de chaque domaine.

* * *

[127] Solution qui semble autoriser le plus de flexibilité vis à vis de la gestion des domaines.

Plan de l'Etape 1

I. Introduction 73
II. POC : une famille de nouveaux protocoles 73
1. Le concept de connexion d'ordre partiel et de fiabilité partielle 73
2. POC : un support adapté au transfert des flux multimédia 74
2.1. Intérêt d'un Transport à fiabilité partielle 74
2.2. Intérêt d'un Transport à ordre partiel 75
3. Proposition d'architecture pour un Transport multimédia 75
3.1. Principes de conception 76
3.2. Impact d'une gestion intégrée de l'ordre et de la fiabilité 77
III. Evaluation des performances 78
IV. Conclusion 80

ETAPE 1 : CONTRIBUTION AUX SERVICES ET PROTOCOLES DE TRANSPORT A QDS

I. Introduction

La première étape de nos travaux a concerné le thème des services et protocoles de Transport pour le transfert de bout en bout des flux multimédia.

Le problème principal que nous avons adressé concernait les limites conceptuelles des protocoles de Transport initiaux, inaptes à tirer partie des caractéristiques nouvelles des applications multimédia (synchronisation et tolérance aux pertes) pour optimiser les performances du service fourni.

Les travaux décrits ci après ont en partie été réalisés dans le cadre du projet CESAME (période de thèse), puis en collaboration avec un doctorant pour l'année après thèse.

Ce chapitre est structuré comme suit : nous résumons en section II le concept proposé pour remédier aux limites des protocoles initiaux ; la section III présente les travaux réalisés en conséquence sur la proposition d'une nouvelle architecture pour les protocoles de Transport multimédia ; enfin, en section IV, nous résumons nos contributions à l'évaluation des performances de l'architecture proposée.

II. POC : une famille de nouveaux protocoles

Tel qu'introduit dans l'état de l'art, les protocoles de Transport initiaux sont fondés sur l'alternative suivante :

- soit ils sont avec connexion (tels que TCP) et offrent des garanties d'ordre et de fiabilité totales sur le transfert des données utilisateur ;
- soit ils sont sans connexion (tels que UDP) et ne garantissent ni ordre ni fiabilité.

Cette classification souligne un vide conceptuel entre les deux familles de protocoles, suggérant une extension du concept de connexion, vis à vis des garanties d'ordre et de fiabilité. Ce nouveau concept, la *connexion à ordre et fiabilité partiels* (POC[128]) a été introduit dans nos travaux [Amer94] [Diaz94] [Chas95].

1. Le concept de connexion d'ordre partiel et de fiabilité partielle

Une POC est une connexion de niveau Transport permettant de définir et de mettre en œuvre tous les services d'ordre partiel et de fiabilité partielle entre deux entités communicantes :

[128] POC : *Partial Order Connection*

– les données échangées peuvent être délivrées au récepteur selon une séquence différente de celle d'envoi par l'émetteur, à condition qu'elle respecte un ordre partiel particulier, échangé en tant que paramètre de QdS lors de l'établissement de la connexion ;
– de façon analogue, ces mêmes données peuvent ne pas être remises au récepteur si les pertes correspondantes sont acceptables par celui-ci ; la fiabilité partielle représente également un paramètre de QdS, échangé lors de l'établissement de la connexion.

Ce nouveau concept étend et unifie les deux approches traditionnelles de la notion de connexion représentées par UDP et TCP, qui apparaissent alors comme deux cas extrêmes de cette proposition (Figure 10).

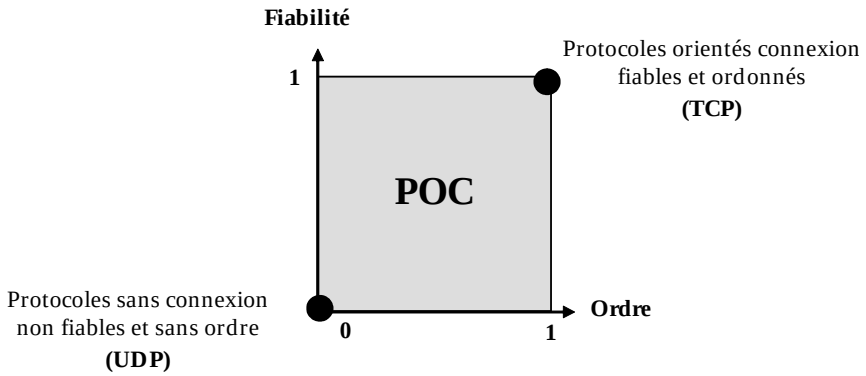


Figure 10 : Espace des protocoles à ordre partiel et à fiabilité partielle

2. POC : un support adapté au transfert des flux multimédia

2.1. Intérêt d'un Transport à fiabilité partielle

L'intérêt d'un Transport offrant des garanties de fiabilité partielle se conçoit aisément. Il permet de prendre en compte, au niveau du réseau, les différences de QdS requise en terme de fiabilité pour le transfert de média discrets (texte) ou continus (audio, vidéo). En cela, il permet une économie de mémoire ou de bande passante, car il diminue le nombre de retransmissions de PDU perdus, et accélère la délivrance des données comparativement à un Transport fiable tel que TCP.

L'expression de la fiabilité partielle peut être envisagée de façon simple, par exemple en termes de taux de pertes ou de nombre maximum de pertes consécutives acceptables. Cependant, la structure des flux audio et vidéo diffère selon les codecs et peut nécessiter des modèles d'expression plus complexes ; c'est par exemple le cas pour les flux H263 ou MPEG qui introduisent des relations de dépendance entre les images I, B, P : de ce fait, le modèle doit tenir compte de la structure des flux et de la sémantique des objets qui le composent.

Nos travaux ont introduit cette dimension [Amer94], en définissant les concepts d'objets *non perdables* (BART_NL[129]), d'objets *perdables* (NBART_L[130]), et d'objets *perdables au delà de N retransmissions* (BART_L(N)[131]). Ces travaux ont ensuite été étendus pour prendre en compte les relations de dépendance (autres que l'ordre) entre les objets [Roja99].

---

[129] BART_NL : *Buffer, Acknowledgement, Retransmission, Time - Not Losable*

[130] NBART_L : *No Buffer, Acknowledgement, Retransmission - Losable*

[131] BART_L(N) : *Buffer, Acknowledgement, Retransmission, Timer - Losable after N retransmissions*

2.2. Intérêt d'un Transport à ordre partiel

L'intérêt d'un Transport offrant des garanties d'ordre partiel est plus subtil à concevoir. Outre des hauts débits et délais bornés, les contraintes des applications multimédia résultent de la structuration des documents manipulés, et s'expriment par les relations de synchronisation entre les différents média (ou flux) qui les composent.

Plusieurs modèles de représentation de la synchronisation multimédia ont été proposés sur la base des réseaux de Petri [Walt83] [Litt90] [Bert91] [Séna94] [Séna96]. Dans tous ces modèles, une place est associée à la présentation d'une donnée[132] ; la structure du réseau et les transitions expriment les relations de dépendance intra-flux et inter-flux.

Le réseau de Petri à la base de tous ces modèles exprime une classe d'ordres partiels, les *ordres partiels série/parallèles*, qui constituent des paramètres potentiels d'une POC.

La Figure 11 illustre le réseau de Petri déduit d'une modélisation TSPN[133] d'un document constitué d'un logo, de 2 images fixes et de 2 vidéos composées chacune de 48 images animées. Cette séquence est supposée se répéter périodiquement. Le parallélisme des places 1, 2 et 3 avec (4-5), …, (98-99) indique que le logo, les 2 images fixes et les 2 vidéos peuvent être présentés indépendamment les uns des autres.

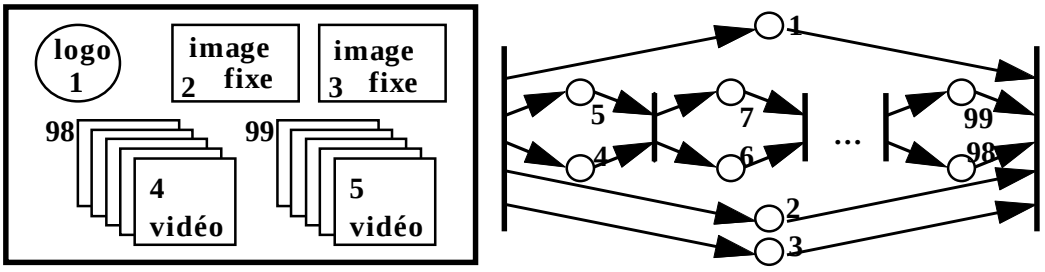


Figure 11 : Réseau de Petri sous-jacent à une modélisation TSPN d'un objet multimédia

Au delà de son intérêt conceptuel, un Transport à ordre partiel permet donc de prendre en compte les caractéristiques de périodicité et de parallélisme des documents multimédia, tout en satisfaisant les contraintes minimales de l'application en termes d'ordre.

En cela, il permet une délivrance des données au plus tôt, ce que ne peut pas faire un Transport totalement ordonné (tel que TCP), et apporte ainsi un élément de réponse aux contraintes de délai des applications multimédia.

Sur la base du concept de POC, nos travaux se sont orientés sur le thème des *architectures de Transport* pour le transfert des flux multimédia.

3. Proposition d'architecture pour un Transport multimédia

Une application multimédia implique le transfert de flux différents (audio, vidéo, etc.), ayant chacun des caractéristiques et des contraintes spécifiques. De ce fait, un service de Transport adapté aux transferts multimédia, doit fournir plusieurs ensembles de QdS, répondant chacun aux besoins d'un flux applicatif.

---

[132] Image, séquence vidéo, échantillon de son, ...

[133] Le modèle TSPN (*Timed Stream Petri Nets*), développé au LAAS-CNRS, permet de spécifier le comportement d'une application multimédia face aux dérives temporelles induites par l'asynchronisme des systèmes [Séna94].

Les travaux de recherche menés dans ce contexte ont conduit à deux types résultats :

- l'extension du concept de QdS Transport [Dant93] [Diaz95], qui s'est traduite par la spécification de nouveaux paramètres de QdS à caractères temporels (délai de transit et gigue) et par la définition de nouvelles sémantiques de service indiquant l'action à entreprendre par le fournisseur du service en cas de dégradation de la QdS négociée ;
- la proposition d'architectures de Transport, répondant aux exigences différentes des flux applicatifs, mais ne prenant pas en compte les contraintes de synchronisation multimédia au niveau du système de communication [Tene94] [Delg93] [Camp94b].

Notre approche d'architecture est différente car elle prend en compte les contraintes de synchronisation multimédia au niveau de la couche Transport ; en cela, elle tire partie des imperfections du réseau pour minimiser les délais de bout en bout, en maximisant les pertes admissibles[134] par l'application.

Nous présentons maintenant les principes de conception d'un Transport multimédia.

3.1. Principes de conception

Une connexion de Transport multimédia à ordre et fiabilité partiels (MM-POC) consiste en l'instanciation puis la *coordination*, au niveau Transport, de plusieurs POC *monomédia*, c'est à dire offrant chacune la QdS requise pour le flux véhiculé (Figure 12).

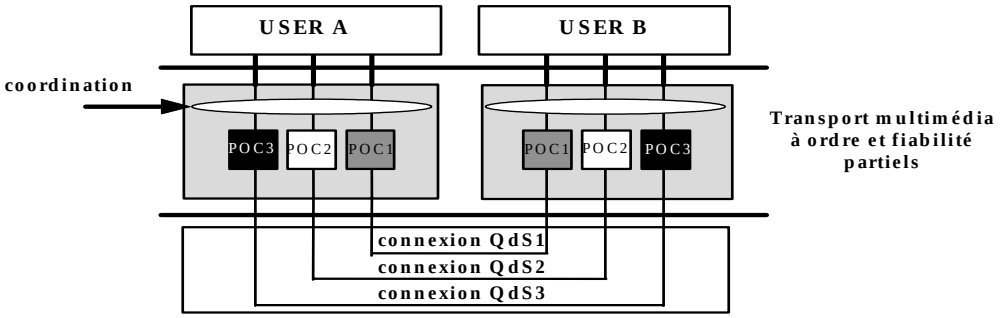


Figure 12 : Exemple d'architecture d'un Transport multimédia à ordre et fiabilité partiels

La coordination est envisagée par le biais de l'ordre partiel et de la fiabilité partielle :

- la gestion de l'ordre partiel se situe non seulement au sein de chaque POC, mais également entre ces POC. En cela, elle permet de prendre en compte les contraintes de synchronisation logique[135] de l'application, à la fois intra et inter-flux ;
- la gestion de la fiabilité partielle peut être envisagée soit *connexion par connexion*, soit *par groupe de connexions* ; dans les deux cas, l'objectif est de permettre la remise au plus tôt d'une donnée non délivrable au regard de l'ordre partiel (intra et inter-flux), au prix de la *déclaration de perte* de la(des) donnée(s) qui la précède(nt) dans l'ordre partiel ; la déclaration de perte doit respecter la fiabilité minimale requise sur chaque POC :
  - pour un contrôle de la fiabilité *connexion par connexion*, les données susceptibles d'être déclarées perdues pour permettre la délivrance d'une donnée *A* appartiennent nécessairement à la même connexion que *A* ;
  - cette restriction disparait pour un contrôle *par groupe de connexions*.

[134] C'est à dire que la fiabilité minimale requise par l'application est garantie.

[135] Par *logique*, nous signifions hors aspects temporels.

La section suivante analyse l'impact d'une gestion intégrée de l'ordre et de la fiabilité au travers de l'architecture proposée.

3.2. Impact d'une gestion intégrée de l'ordre et de la fiabilité

3.2.1. Gestion de l'ordre partiel

Considérons l'ordre partiel *P* représenté Figure 13, déduit du modèle TSPN d'une application manipulant trois flux : audio, vidéo 1 et vidéo 2.

P fait apparaître deux types de relations de dépendance :

- les relations de dépendance *intra-flux*, telle que « l'objet 4 précède les objets 5 et 6 dans l'ordre partiel du flux audio » ;
- les relations de dépendance *inter-flux*, telle que « l'objet 2 précède les objets 8 et 9 dans l'ordre partiel du flux multimédia ».

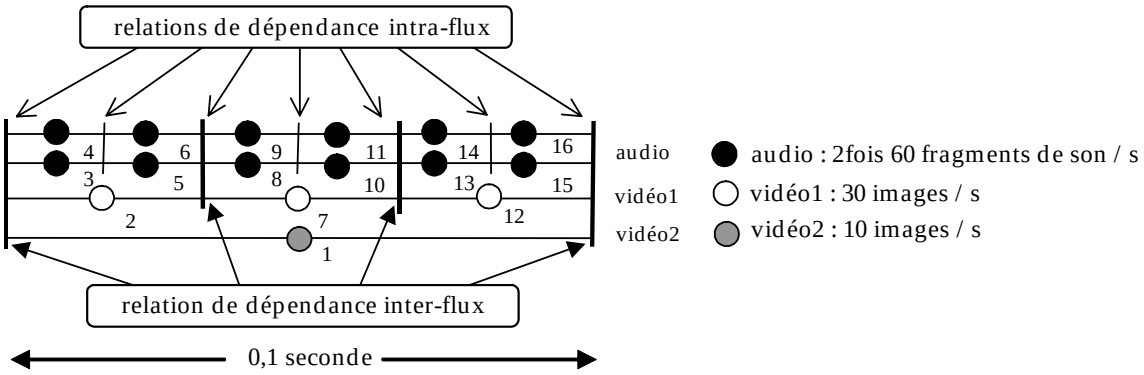


Figure 13 : Ordre partiel multimédia P et relations de dépendance intra et inter-flux

La prise en compte de *P* dans l'architecture de Transport multimédia de la Figure 12 se traduit à deux niveaux :

- d'une part au sein de chaque POC, pour respecter les relations de dépendances intra-flux du flux véhiculé ; par exemple, les relations de dépendance propres au flux audio imposent à la POC 3, de délivrer les objets 3 et 4 avant les objets 5 et 6 ;
- d'autre part entre les POC, pour respecter les relations de dépendances inter-flux. Par exemple, la délivrance des objets audio 8 et 9 ne doit intervenir qu'après celle de l'objet 2, du flux vidéo1.

3.2.2. Gestion de la fiabilité partielle

A l'échelle d'une connexion *monomédia* (POC), l'intégration de la fiabilité partielle se traduit par une politique de délivrance au plus tôt d'un PDU reçu, au prix de la déclaration de perte (acceptable) des PDU le précédant dans l'ordre partiel intra flux, que ceux-ci aient été effectivement perdus ou qu'ils soient retardés.

Il s'agit donc de contrôler, lors de la délivrance d'un PDU, la dégradation de la fiabilité consécutive à une éventuelle déclaration de perte en appliquant la règle suivante :

« *la délivrance d'un PDU rend obsolète tous les PDU non encore délivrés ni déclarés perdus, qui le précédent dans l'ordre partiel* ».

Dans l'exemple d'ordre partiel de la Figure 14, supposons le PDU 4 délivré et le PDU 5 parvenu à l'entité réceptrice tandis que le PDU 3 n'est pas encore reçu.

Si le PDU 3 peut être déclaré perdu (au regard de la fiabilité partielle requise), le PDU 5 sera alors délivré à l'utilisateur sans délai.

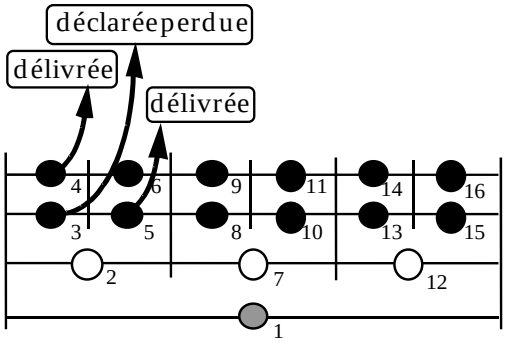


Figure 14 : Exemple d'ordre partiel

A l'échelle de la multi connexion, le contrôle de la fiabilité est envisageable de deux façons :

- soit c*onnexion par connexion*, ce qui impose que la délivrance d'un PDU sur une POC_i n'affecte pas la fiabilité d'une POC_j avec $i \neq j$;
- soit *par groupe de connexions*, ce qui autorise la délivrance d'un PDU sur une POC_i même si elle nécessite la déclaration de perte de PDU(s) d'une ou plusieurs POC_j avec $i \neq j$.

Supposons le PDU 3 déclaré perdu et les objets 4 et 5 délivrés à l'utilisateur (Figure 14). Si à présent le PDU 8 parvient à l'entité de Transport réceptrice, celui ci est :

- stocké jusqu'à réception ou perte avérée[136] du PDU 2 dans le cas d'une gestion *connexion par connexion*, le PDU 6 étant alors déclaré perdu[137] ;
- délivré sans attente à l'application tandis que les objets 2 et 6 sont déclarés perdus avec un mécanisme de gestion *par groupe de connexions*.

Ce scénario illustre comment l'architecture proposée permet de tirer partie d'une dégradation acceptable de la fiabilité pour délivrer les PDU au plus tôt, au prix d'une déclaration de perte du(des) PDU(s) retardé(s) ou effectivement perdu(s) par le réseau.

Dans les deux cas, le contrôle de la fiabilité contribue à diminuer le délai de transit des données tout en garantissant les contraintes d'ordre intra et inter flux et de fiabilité pour chaque flux. Les deux options envisagées, de complexité équivalente, diffèrent par la priorité donnée aux performances (délai de transit) dans le cas d'un contrôle *par groupe de connexions* ou à l'indépendance entre les flux dans le cas d'un contrôle *connexion par connexion*.

III. Evaluation des performances

Dans le cadre des travaux de thèse décrits dans [Four97a], nous avons contribué à l'évaluation des performances de deux modèles OPNET de protocoles à ordre et fiabilité partiels : l'un monomédia (POC) [Four97b], l'autre multi média (MM-POC) [Four99c].

---

[136] Cette perte pourra être avérée sur réception de l'objet 7.

[137] Les objets déclarés perdus ne sont en aucun cas délivrés même si ils parviennent avec retard à l'entité Transport réceptrice.

Dans [Four99c], l'objectif était d'évaluer l'impact moyen et maximal des deux types de gestion sur l'utilisation des ressources mémoire (en émission et en réception) et sur le délai de transit. Les paramètres variant étaient le taux de perte du réseau et la fiabilité requise par l'application, identique sur chaque POC dans le cas de la gestion multimédia. Quatre valeurs ont été considérées pour le taux de pertes réseau : 0, 5, 10 et 25%, et six valeurs ont été considérées pour la fiabilité 0.5, 0.6, 0.7, 0.8, 0.9 et 1 signifiant respectivement que l'utilisateur tolère 50, 40, 30, 20, 10 et 0% de pertes. L'ordre partiel et le délai nominal du réseau était fixés (Figure 15).

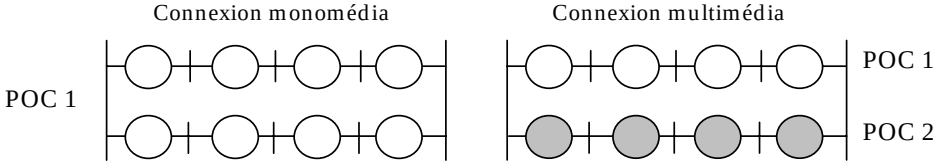


Figure 15 : Ordre partiel

Nous n'illustrons ici qu'une partie des résultats obtenus sur le délai de transit. Les conclusions sont identiques pour les ressources mémoires.

La Figure 16 illustre les gains moyens induits sur le délai de transit. On constate que le délai de transit diminue avec la fiabilité jusqu'à devenir minimal lorsque celle-ci est inférieure au taux de pertes du réseau. Ce résultat est logique et confirme l'intérêt d'une prise en compte, au niveau Transport, de la tolérance aux pertes des applications multimédia.

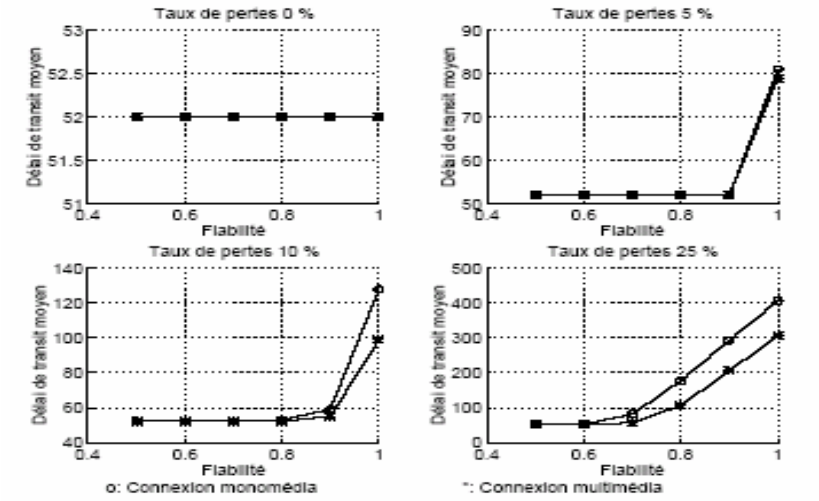


Figure 16 : Résultat obtenus pour le délai de transit moyen

Dans tous les cas, ces résultats illustrent également l'intérêt d'utiliser une connexion de Transport multimédia plutôt que monomédia. Pour une fiabilité requise supérieure au taux de perte du réseau, des gains allant jusqu'à 25 % sont ainsi mis en évidence.

Ce constat est lié au degré de parallélisme des traitements qu'induisent les solutions POC et MM-POC envisagées. En traitant les flux applicatifs par le biais de connexions distinctes, une MM-POC autorise un plus haut degré de parallélisme qui se traduit par une mémorisation moins longue des PDU et un délai de transit applicatif en conséquence réduit.

IV. Conclusion

Les travaux que nous avons présentés dans cette étape visaient à repousser les limites conceptuelles des protocoles de Transport vis à vis des caractéristiques et des besoins des applications multimédia.

L'objectif était d'optimiser, d'une part les performances du service offert (délai de transit) et d'autre part, les ressources du système de communication (bande passante et buffer de stockage au niveau des hôtes). Pour cela :

- nous avons tout d'abord proposé un nouveau concept de services et protocoles, à ordre et fiabilité partiels, permettant de prendre de compte la tolérance aux pertes/erreurs et les contraintes de synchronisation des applications multimédia, pour combler les imperfections du réseau en terme de pertes et de déséquencements ;
- dans un second temps, nous avons proposé une nouvelle architecture, à multi connexions, pour les protocoles de Transport multimédia, permettant d'étendre les gains induits par une gestion de l'ordre et de la fiabilité partiels, comparativement à un Transport monomédia (c'est à dire mono connexion) ;
- enfin, nous avons contribué à l'évaluation des performances de ces propositions de nouveaux protocoles et architectures, pour en démontrer le bien fondé.

Cependant, les avancées effectuées au niveau de l'amélioration du Transport atteignent leurs limites lorsque le service fourni par le réseau ne permet d'assurer ni un délai borné ni un taux de perte borné sur le transfert des données.

En ce sens, nos préoccupations ont considéré les travaux de la communauté Internet visant à améliorer la QdS de celui ci au niveau IP. Nous présentons ces travaux ci-après dans l'Etape 2 de notre parcours de recherche.

* * *

Plan de l'Etape 2

I. Introduction 83
II. Caractérisation de la QdS requise par les applications DIS 83
1. Introduction à la simulation interactive distribuée 83
1.1. Généralités 83
1.2. Caractéristiques du protocole DIS 84
2. QdS requise par une application DIS 85
2.1. Limites du standard DIS 85
2.2. Proposition de QdS 87
III. Caractérisation du trafic DIS 90
1. Principe 90
2. Validation expérimentale 91
IV. Architecture DIS/ATM 91
1. Principes 92
1.1. DIS+ 92
1.2. RSVM 93
2. Déploiement et mesures de performance 93
V. Conclusion 94

ETAPE 2 : CONTRIBUTION AUX ARCHITECTURES A QDS EN CONTEXTE INTSERV

I. Introduction

La première approche de solution pour l'obtention d'une QdS IP dans l'Internet résulte des travaux du groupe IntServ de l'IETF.

Comme mentionné dans l'état de l'art, IntServ laisse ouvert l'interfaçage des applications avec le système proposé, via une API dont les paramètres de spécification du trafic et de caractérisation de la QdS requise sont clairement orientés réseau [Brad98].

C'est dans ce contexte que nous avons abordé le thème des architectures pour la QdS de bout en bout, pour une famille d'applications typiques, la simulation interactive distribuée (DIS), impliquant l'échange d'un trafic à multiples facettes difficilement caractérisables, tant vis à vis du débit généré que de leurs besoins en QdS.

L'objectif était de définir comment offrir une QdS compatible avec les besoins DIS, en déchargeant l'utilisateur du choix et du paramétrage des services IntServ sous jacents.

Les deux problématiques que nous avons adressées étaient :

- de caractériser le trafic et les besoins en QdS des applications DIS ;
- de définir une architecture de bout en bout effectuant, de façon transparente pour l'application, le choix, le paramétrage et l'activation des services IntServ sous jacents.

Les travaux correspondant ont été réalisés dans le cadre du projet DIS/ATM.

La suite de ce chapitre est structurée comme suit. Les sections 1 et 2 présentent nos contributions à la caractérisation de la QdS requise par DIS, puis du trafic émis par DIS. Nos contributions à la définition, l'implémentation et l'évaluation des performances d'une architecture répondant la deuxième problématique sont présentés en section 3.

II. Caractérisation de la QdS requise par les applications DIS

1. Introduction à la simulation interactive distribuée

1.1. Généralités

La simulation interactive distribuée est issue du domaine militaire ; elle avait pour premier objectif de préparer, en simulation, des unités à toute forme d'engagement dans des environnements divers.

Un *exercice DIS* regroupe plusieurs sites interconnectés en réseau :

- sur chaque site s'exécutent une ou plusieurs applications, partageant un même espace virtuel (terrestre, maritime ou aérien) ;
- chaque application simule le comportement, d'une part d'entités[138] locales, autonomes ou sous la conduite d'acteur(s) humain(s), et d'autre part, d'entités distantes, à partir d'informations régulièrement fournies par les applications qui les gouvernent ;
- pour rendre interopérables toutes les applications d'un même exercice, le protocole de niveau applicatif DIS [Sdt 1278.1] [Sdt 1278.2] définit le format et les règles d'échange des informations transmises entre applications de simulation distantes (Figure 17).

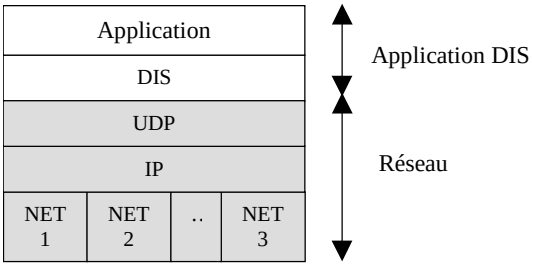


Figure 17 : Architecture de base d'une application DIS (en réseau local)

1.2. Caractéristiques du protocole DIS

Le standard DIS définit 27 PDU DIS regroupés en 6 familles parmi lesquelles :

- la famille *Entity Information/Interaction,* formée de 2 PDU, *Entity State* (ES) véhiculant l'état[139] d'une entité et *Collision* ; les PDU ES constituent environ 96% du trafic total ;
- la famille *Warfare* composée de 2 PDU, *Fire* et *Detonation,* liés aux tirs de projectiles ;
- la famille *Simulation Management* regroupant 12 PDU de gestion de la simulation (entrée ou sortie d'exercice, création / destruction d'entités, etc.) ;
- la famille *Radio Communication* (3 PDU) répondant aux besoins en échanges entre participants, en simulation de détection électronique.

Le maintien de l'état d'une entité repose sur la diffusion d'un PDU ES par le site qui la gouverne et sur l'actualisation de l'état de cette entité sur tous les autres sites. La procédure est basée sur la mise en œuvre d'un algorithme de *Dead Reckoning* (DR), dont l'objectif est :

- d'une part, de diminuer le trafic en réduisant l'émission des PDU ES ;
- d'autre part, de permettre aux sites récepteurs *d'extrapoler* l'état d'une entité, à partir des informations contenues dans les PDU *ES* déjà reçus.

Les principes du DR sont les suivants :

- pour chaque entité gouvernée, un site maintient les représentations réelle et extrapolée de l'état de cette entité. Il émet un PDU *ES* si l'une des conditions suivantes est vérifiée :
  - l'écart entre l'état extrapolé et l'état réel excède l'un des seuils définis dans le standard pour la position ou pour l'orientation ;

[138] Chars, avions, …

[139] Position, vitesse, accélération.

- l'état de l'entité n'a pas été rafraîchi depuis un temps égal au *HEART BEAT TIMER*, fixé à cinq secondes par défaut dans le standard ;

– sur les autres sites, la réception à l'instant t_0 d'un PDU *ES,* provoque la mise à jour de l'état de l'entité concernée ; il est possible d'estimer sa position $\mathbf{P}_{DR}(t)$ à tout instant $t > t_0$, grâce à une extrapolation linéaire[140] (relation 1) ou quadratique[141] (relation 2).

$$\mathbf{P}_{DR}(t) = \mathbf{P}_0 + \mathbf{V}_0.(t - t_0) \qquad [1]$$

$$\mathbf{P}_{DR}(t) = \mathbf{P}_0 + \mathbf{V}_0.(t - t_0) + \frac{1}{2}\mathbf{A}_0.(t - t_0)^2 \qquad [2]$$

2. QdS requise par une application DIS

Une application DIS fonctionne correctement si les différents sites ont une vision *cohérente* de la simulation. Cette cohérence revêt deux aspects, un aspect *temporel* relatif aux événements et un aspect *spatial* relatif aux états des entités :

– la *cohérence temporelle* signifie que tout site doit connaître l'occurrence de tout événement intervenu sur un autre site avec un retard borné ;

– la *cohérence spatiale* signifie que tout site doit connaître à chaque instant l'état de toute entité du système, « à une certaine erreur près ».

Pour illustrer ce deuxième aspect, considérons une entité *A* (ici un avion), contrôlée par le site S_1. La Figure 18 illustre sur un même plan les positions et orientations réelles de l'entité *A* sur son site de simulation S_1 et leur extrapolation sur le site S_2.

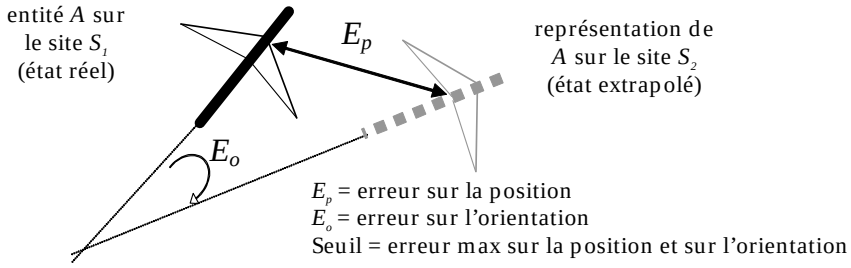


Figure 18 : Position et orientation d'une entité

La cohérence spatiale impose qu'à tout instant, l'écart entre l'état attribué à *A* sur le site S_2 et l'état réel de *A* n'excède pas un certain seuil, défini dans le standard DIS comme l'erreur maximale acceptable sur la position, Th_{Pos}, et sur l'orientation, Th_{Or}.

S_1 et S_2 ont une vue spatialement cohérente de *A* tant que $|E_p| \leq Th_{Pos}$, et $|E_o| \leq Th_{Or}$.

2.1. Limites du standard DIS

Dans le standard DIS, les besoins en QdS des applications DIS sont exprimés en termes de taux de pertes maximum (τ), de délai de transit maximum, (DT) et de gigue (pour l'audio). Les valeurs de τ et DT dépendent du *couplage* des entités auxquelles sont associés les PDU, que le standard DIS illustre ainsi :

---

[140] Si l'on ne dispose que de la position $\mathbf{P}_0$ et de la vitesse $\mathbf{V}_0$ de l'entité à l'instant t_0

[141] Si l'on dispose en outre de l'accélération $\mathbf{A}_0$ à l'instant t_0

– deux chars distants de 5 kilomètres, sont des entités *faiblement couplées* ;
– plusieurs chars en formation évoluant rapidement sont des entités *fortement couplés*.

Sur ces bases, les valeurs des paramètres τ et DT proposées par le standard sont :
– $DT \leq 100$ ms et $\tau \leq 2$ % dans le cas d'un couplage fort ;
– $DT \leq 300$ ms et $\tau \leq 5$ % dans le cas d'un couplage faible.

Cette expression de la QdS présente deux limites principales :
– la première tient à la notion de couplage qui impose :
  • d'évaluer, lors de l'émission d'un PDU ES d'une entité donnée, le degré de couplage entre celle-ci et toute entité distante ;
  • partant de cette évaluation, de déterminer la QdS requise pour le transfert de ce PDU vers chacun des sites ;
– la seconde est que le standard ignore l'influence du délai de transit sur l'erreur de position / orientation, négligeant en cela les contraintes de cohérence spatiale.

La première limite se manifeste par le temps de calcul nécessaire à l'évaluation du couplage entre entités, et le recours à une paramétrisation très dynamique (PDU par PDU) de la qualité du service sous jacent. Pour repousser cette limite, une solution consisterait à traiter tous les PDU ES comme s'ils émanaient d'entités fortement couplées, ce qui aboutirait à une sur-utilisation des ressources, sans pour autant répondre à la deuxième limite.

Cette deuxième limite est liée au respect de la cohérence spatiale. Compte tenu du délai de transit des PDU ES, la mise à jour par DR de l'état des entités est différée sur les sites distants vis à vis du site émetteur. En conséquence, durant l'intervalle de temps qui sépare l'émission d'un PDU ES de sa réception sur le site distant, une violation de la cohérence spatiale peut apparaître. Nous illustrons ci-après les conditions d'apparition de ce phénomène.

Soient deux sites S_e (émetteur) et S_r (récepteur) participant à un même exercice DIS. On s'intéresse à une entité A simulée sur S_e, dont l'état est mis à jour sur S_r par application d'un algorithme de DR. Pour simplifier la présentation, l'état de A sera réduit à la seule position[142] que l'on supposera varier selon une seule dimension (mouvement rectiligne).

La Figure 19 illustre l'évolution, en fonction du temps, des erreurs d'extrapolation de la position de A, E_e et E_r, faites respectivement sur le site S_e et sur le site S_r[143] :
– les pastilles noires marquent l'émission des PDU ES rafraîchissant l'état de A. On choisit pour origine des temps une date d'émission (T_{e0}) ;
– les pastilles grises indiquent les dates de réception des PDU ES. Les dates T_{r0}, T_{r1}, T_{r2} et T_{r3} sont les dates de réception des PDU ES émis aux dates respectives T_{e0}, T_{e1}, T_{e2} et T_{e3}.

Analysons les dates T_{e1}, T_{e2} et T_{e3} :
– à partir de T_{e0}, l'erreur d'extrapolation de la position E_e augmente et atteint la valeur maximale admissible (Th_{Pos}, ou seuil de DR sur la position) à la date T_{e1} : le DR provoque alors l'émission d'un PDU ES contenant la position réelle de A ; l'erreur E_e s'annule ;
– à partir de T_{e1}, le scénario précédant se reproduit jusqu'à la date T_{e2} ;

[142] Un raisonnement analogue peut être appliqué à l'orientation à l'aide du vecteur d'Euler.

[143] Seul le site émetteur S_e est en mesure de calculer effectivement l'erreur d'extrapolation.

– à partir de T_{e2}, l'erreur oscille entre - Th_{Pos} et + Th_{Pos} sans jamais quitter l'intervalle autorisé ; au bout du HEART BEAT TIMER, soit à $T_{e3} = T_{e2} + 5$ secondes, le DR émet un PDU ES et l'erreur E_e redevient nulle.

Analysons à présent l'évolution de l'erreur E_r faite sur le site S_r :

– à partir de T_{r0}, l'erreur de la position E_r est identique à celle faite sur S_e ; elle atteint la valeur maximale admissible Th_{Pos} à T_{e1}, date d'émission du PDU ES rafraîchissant A. Cependant, la mise à jour de la position de A n'est effective qu'à la date T_{r1}, l'intervalle séparant T_{e1} et T_{r1} correspondant au délai de transit du PDU ES. Il apparaît donc une indétermination sur E_r entre les dates T_{e1} et T_{r1}, durant laquelle E_r peut excéder la valeur Th_{Pos}, engendrant par là même une violation de la cohérence spatiale ;
– de manière générale, il apparaît que E_r n'est pas maîtrisée durant les intervalles de temps matérialisés en gris sur la Figure 19.

Ce phénomène est préjudiciable dès lors que le délai de transit n'est plus négligeable devant l'intervalle de temps séparant deux réceptions consécutives de PDU *ES*. Ce risque, masqué en réseau local du fait des très faibles valeurs des délais de transit observés, est important dans un réseau à grande distance, ce qui était typiquement le contexte de notre étude.

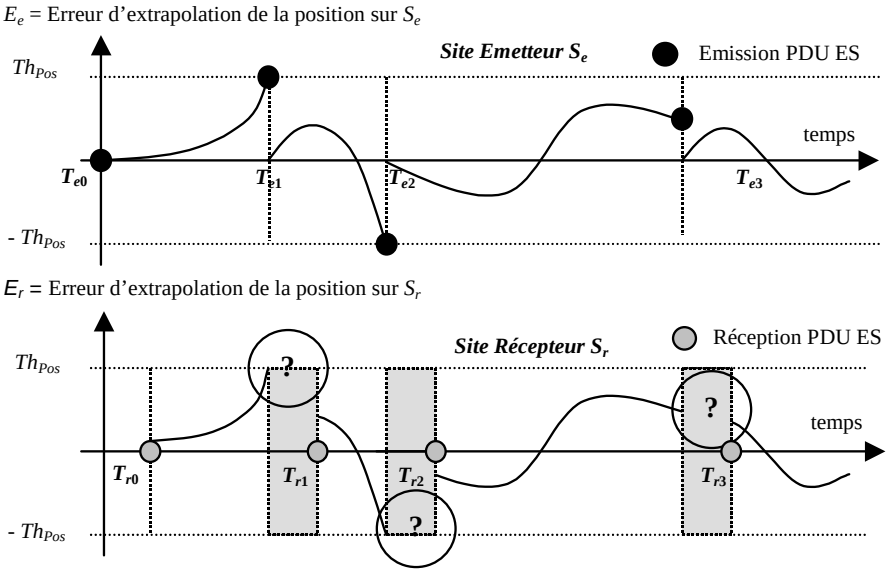


Figure 19: Evolution de l'erreur d'extrapolation côté émetteur et côté récepteur

Nous présentons maintenant notre proposition qui s'affranchit des limites du standard en délaissant la notion de couplage, mais en garantissant la cohérence spatiale.

2.2. Proposition de QdS

La QdS associée au transport des PDU DIS est exprimée au moyen des trois mêmes paramètres que ceux définis dans le standard[144], mais :

– la notion de couplage est abandonnée ;
– la valeur des paramètres précédents est déterminée :

[144] Taux maximum de pertes, délai de transit maximum et gigue.

- à partir des seules informations contenues dans les PDU ES ; en cela, notre proposition remédie au surcoût en temps de calcul et à la détermination de la QdS de chaque PDU ;
- pour garantir à tout instant une erreur maximale, en particulier durant les périodes transitoires précédemment décrites ; en cela, notre proposition répond aux contraintes de cohérence spatiale.

2.2.1. Mise en œuvre de la garantie d'une erreur maximale

Pour garantir une borne supérieure E_{Tmax} à l'excès transitoire[145] E_T, il est nécessaire que le délai de transit DT des PDU ES n'excède pas une valeur DT_{max} (Figure 20). Intuitivement, l'excès transitoire E_T dépend de la dynamique (accélération) de l'entité simulée ; en conséquence, DT_{max} dépend également de la dynamique de l'entité.

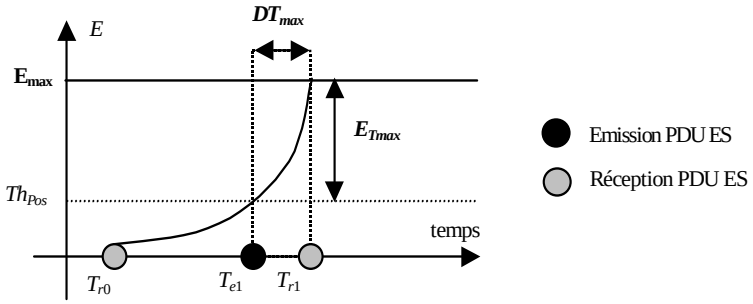


Figure 20: Garantie d'un excès transitoire maximal

L'influence du délai de transit DT sur la connaissance de l'état des entités par chaque site est liée à l'expression de l'écart en valeur absolue $e_{\mathbf{P}}(t)$, entre la position effective $\mathbf{P}_a(t)$ et la position extrapolée $\mathbf{P}_{DR}(t)$ pour une entité quelconque (relation 3) :

$$\begin{cases} \mathbf{P}_a(t) = \int_{Te_i}^{t} du \int_{Te_i}^{u} \mathbf{A}_a(\tau) \cdot d\tau + (t - Te_i)\mathbf{V}_i + \mathbf{P}_i \\ e_{\mathbf{p}}(t) = \left\| \mathbf{P}_a(t) - \mathbf{P}_{DR}(t) \right\| = \left\| \int_{Te_i}^{t} du \int_{Te_i}^{u} \left[\mathbf{A}_a(\tau) - \mathbf{A}_i \right] \cdot d\tau \right\| \end{cases} \quad [3]$$

où :

- $\mathbf{A}_a(t)$ représente l'accélération effective de l'entité à l'instant t,
- $\mathbf{P}_i$, $\mathbf{V}_i$, et $\mathbf{A}_i$ représentent respectivement la position, la vitesse et l'accélération de l'entité au temps Te_i.

Notons que $\mathbf{P}_i$, $\mathbf{V}_i$, et $\mathbf{A}_i$ sont trois champs du PDU *ES* ($\mathbf{A}_i$ vaut 0 pour la relation (1) du *DR*).

Sous l'hypothèse que l'entité possède une accélération bornée $A_{\max}$, $\|\mathbf{A}_a(t)\| \leq A_{\max}$, $e_{\mathbf{P}}(t)$ est majoré dans l'intervalle de temps $[Te_{i+1}, Te_{i+1}+DT]$, et par suite, on peut majorer $e_{\mathbf{P}}(Te_{i+1}+DT)$.

[Chas00] montre que cette majoration et bornée supérieurement dans les cas d'une extrapolation linéaire ou quadratique ; pour que l'erreur transitoire E_T soit inférieure à une valeur maximale E_{Tmax}, il faut garantir pour chaque PDU ES un délai de transit DT_{max} dont la valeur est fonction de l'accélération maximale de l'entité associée (relation 4) :

[145] E_T est défini comme la différence entre la valeur absolue de l'erreur d'extrapolation sur un site distant pour t > Te_i et le seuil Th_{pos} comme indiqué Figure 20.

$$DT_{max} \leq \begin{cases} \dfrac{\sqrt{Th_{Pos} + E_{Tmax}} - \sqrt{Th_{Pos}}}{\sqrt{0{,}5 \cdot A_{max}}} & \text{extrapolation linéaire} \\ \dfrac{\sqrt{Th_{Pos} + E_{Tmax}} - \sqrt{Th_{Pos}}}{\sqrt{A_{max}}} & \text{extrapolation quadratique} \end{cases} \qquad [4]$$

Ainsi, il est possible d'assurer la cohérence spatiale en garantissant la position de toute entité à l'intérieur d'une boule d'incertitude de rayon $Th_{Pos} + E_{Tmax}$, à condition de garantir un délai de transit des PDU *ES* inférieur ou égal à DT_{max}[146].

La section suivante montre comment répartir les PDU DIS en classes de QdS, en précisant la sémantique et la valeur des paramètres de chacune de ces classes.

2.2.2. Classification des PDU DIS

La répartition proposée comporte quatre classes :

- la classe des PDU événementiels qui requiert :
  - une fiabilité totale, jugée nécessaire du fait de l'impossibilité de récupérer l'information attachée aux PDU concernés en cas de perte ;
  - un délai de transit maximal, DT_{max}, de 100 ms ; compte tenu de la sémantique des PDU concernés, la valeur choisie correspond au cas du couplage fort dans le standard DIS.
- la classe des PDU de gestion de la simulation, qui requiert :
  - une fiabilité totale, le standard DIS ne précisant pas l'action à entreprendre en cas de perte d'un PDU de cette classe ;
  - un DT_{max} non borné, ces PDU ne présentant pas de contrainte temporelle spécifique.
- la classe du PDU Signal, qui requiert :
  - une fiabilité « élevée », traduisant le fait qu'un transfert audio peut présenter diverses contraintes de fiabilité selon le codage utilisé ;
  - un DT_{max} de 250 ms, délai habituel pour une application d'audioconférence ;
  - une gigue maximale de 50 ms, valeur retenue dans les applications d'audioconférence ;
- la classe des PDU continus (*Entity State* notamment qui constituent la très grande majorité du trafic), qui requiert :
  - une fiabilité totale ;
  - un DT_{max} fonction de l'accélération de l'entité ; en conséquence sont définies plusieurs sous-classes de cette classe présentées dans le tableau de la Figure 21.

A chaque plage d'accélération maximale $[A_{i-1}, A_i]$ est associée un délai de transit DT_i dont la valeur est déduite de la formule (6) qui garantit le respect d'une erreur transitoire maximale E_{Tmax} pour les entités dont l'accélération maximale est comprise entre A_{i-1} et A_i.

---

146 Remarquons que pour une même précision souhaitée, $Th_{Pos} + E_{Tmax}$, on peut faire varier la valeur de DT_{max}, mais l'abaissement des contraintes sur le délai de transit conduit en moyenne à une augmentation du débit : (a) si l'on diminue le seuil Th_{Pos}, DT_{max} augmente et les contraintes temporelles sont plus faciles à satisfaire ; en contre partie, la mise à jour des PDU *ES* risque d'être plus fréquente, ce qui augmente le débit ; (b) lorsque l'on passe d'une interpolation linéaire à une interpolation quadratique, les contraintes sur DT_{max} augmentent, mais la fréquence moyenne de mise à jour des PDU devrait diminuer, l'extrapolation étant améliorée par la prise en compte de l'accélération.

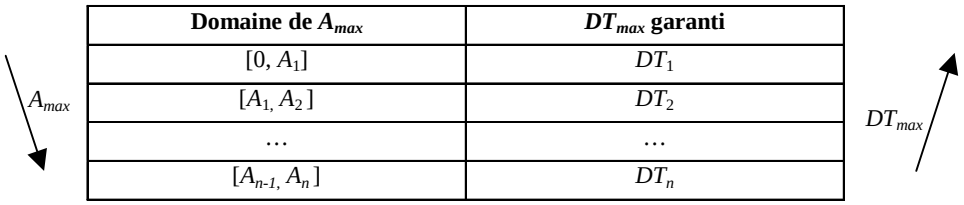


Figure 21: Répartition des entités en sous-classes de QdS

III. Caractérisation du trafic DIS

Dans les réseaux à QdS, le contrat négocié entre l'application et le réseau implique que l'application respecte en émission le profil de trafic annoncé. Avant tout engagement, l'application doit donc estimer le profil du trafic qu'elle est susceptible d'engendrer.

Dans le cas d'IntServ, le modèle de caractérisation du trafic est le TSPEC[147] de RSVP, qui inclut en particulier comme paramètres le débit moyen (*r*), le débit crête (*p*), et la taille maximale des rafales d'octets de paquets IP (*b*)[148]. Les paramètres du TSPEC étant orientés réseau, leur spécification n'est pas évidente du point de vue de l'application.

Afin de résoudre ce problème, nous avons proposé une méthodologie qui relie les paramètres pertinents au niveau applicatif (DIS) et ceux du TSPEC, en particulier les paramètres *r*, *b* et *p*.

1. Principe

Soit un trafic donné *T* caractérisé par les paramètres du TSPEC de RSVP.

La méthodologie proposée repose sur l'hypothèse générale que *T* résulte de l'agrégation de trafics individuels engendrés par des éléments applicatifs (EA) distincts et identifiables.

De façon plus précise, cette hypothèse exprime que :

- l'application est décomposable en EA générant chacune leur propre trafic ; pour DIS, nous avons choisi de faire coïncider, au niveau d'un site, les notions d'EA et d'entités de même *morphologie*[149] simulées localement ;
- le trafic d'une EA doit être (statistiquement) reproductible dans le temps et similaire pour deux représentants d'une même EA ; dans le cas de DIS, nous avons cherché à vérifier si cette hypothèse était satisfaite en analysant le trafic d'un exercice DIS pré-enregistré.

Sous ces conditions, la prédiction du TSPEC correspondant à l'agrégation des trafics des EA est définie par la somme des paramètres *r*, *b*, *p* de chacun des trafics individuels générés par un représentant de chaque EA.

Notons que l'addition constitue la loi de composition exacte pour *r*, mais une majoration *p* et b. Là encore, nous avons vérifié la validité de la méthode en comparant le profil de trafic estimé avec le profil de trafic réel de l'exercice pré-enregistré.

---

147 TSPEC : *Traffic Specification*

148 Ces paramètres sont utilisés dans le modèle du *token bucket*.

149 Le standard DIS définit la *morphologie* d'une entité par un certain nombre de critères (que l'on retrouve dans l'entête des PDU) décrivant l'aspect physique et le comportement de l'entité.

2. Validation expérimentale

La validation expérimentale est basée sur l'utilisation d'un logiciel gratuit qui capture les PDU produits lors d'un exercice DIS, puis les rejoue en respectant l'ordre et le rythme indiqués par leur date de capture.

Nous avons travaillé sur un scénario pré-enregistré afin de vérifier :

- l'hypothèse de similarité du trafic généré par deux entités d'une même morphologie ;
- le bien fondé de la méthode de prédiction du trafic agrégé.

Compte tenu du manque de contrôle que nous possédions sur l'enregistrement DIS initial, nous avons tout d'abord procédé à un filtrage des entités et des sites qui donnaient lieu à un nombre insuffisant de PDU ou à une trop grande variance sur le délai[150].

Nous avons retenu pour cette validation six morphologies, pour lesquelles le nombre d'entités étaient supérieur ou égale à cinq.

La première expérimentation visait à vérifier l'hypothèse selon laquelle des entités d'une même morphologie produisent un trafic statistiquement comparable. Pour cela, nous avons enregistré, pour chacun des paramètres du TSPEC, les valeurs minimale, maximale et moyenne observées sur l'ensemble des entités de la même morphologie. Ces résultats ont montré le peu de variance de comportement entre les entités d'un même type, en particulier pour les paramètres *r* et *b*.

La deuxième expérimentation avait pour but de vérifier la validité de la prédiction de trafic suivant la méthode proposée :

- d'une part, en choisissant, pour chaque morphologie, le trafic d'une entité prise au hasard, et en composant le TSPEC (par addition) pour estimer le profil de trafic correspondant au nombre total d'entités de la même morphologie ;
- d'autre part, en déterminant pour chaque morphologie, le profil de trafic effectif correspondant aux entités.

La comparaison entre le TSPEC prédit et le profil de trafic observé (Tableau 2) a donné des résultats satisfaisants pour le débit moyen *r* et la taille des rafales *b*, mais pas pour le débit crête *p*, ce qui était prévisible, du fait de la dispersion des instants d'émission.

Morphologie		Nombre	*r* (bits/s)	*b* (bits)	*p* (bits/s)
1	Missile anti navire guidé	16	171	16670	115200
2	Véhicule blindé (UK)	11	967	179700	422400
3	Autre véhicule blindé	6	664	88790	140800
4	Artillerie motorisée	5	154	1144	115200
5	Tank (U.K.)	17	706	231700	281600
6	Autre tank	11	345	4248	140800
TSPEC prédit (total)			33924	6767508	14246400
TSPEC observé			30130	5528000	3520000

Tableau 2 : TSPEC estimé v.s. TSPEC observé

IV. Architecture DIS/ATM

[150] En raison de problème d'horloge locale ou de délai trop variable entre le site émetteur et l'enregistreur/joueur.

Nous présentons maintenant l'architecture de bout en bout définie pour satisfaire les besoins des applications DIS dans un contexte Internet de type IntServ.

1. Principes

Les objectifs de conception de l'architecture étaient :

– d'une part, de satisfaire la QdS requise en minimisant l'utilisation des ressources réseau ;

– d'autre part, d'abstraire l'application de la complexité d'utilisation de l'API RSVP.

Vis à vis du premier objectif, notre proposition repose sur l'établissement d'un canal de bout à bout pour chacune des classes de PDU DIS, chaque canal offrant une QdS spécifique à celle requise pour la classe considérée. Notons que ce choix rejoint celui adopté pour les flux multimédia dans les propositions d'architecture [Camp94a], [Gopa95], [Nahr96], [Chas96].

Vis à vis du deuxième objectif, notre proposition repose sur un gestionnaire de QdS, dont le rôle est de réserver les ressources pour chaque canal, à partir d'une caractérisation du trafic DIS en des termes compréhensibles par l'application ou un utilisateur averti (via une IHM).

L'architecture proposée distingue ainsi deux plans (Figure 22) :

– un plan de transfert de données multi canaux (DIS+) s'appuyant sur UDP, chargé :
  - d'établir les canaux de bout en bout ;
  - d'aiguiller les PDU DIS sur chacun d'eux, en fonction de leur classe d'appartenance ;

– un plan de gestion de la QdS (RSVM) s'appuyant sur RSVP, ayant pour fonction, à partir des paramètres de caractérisation du trafic exprimés via une API (RMAPI) :
  - de déterminer le profil de trafic (TSPEC) pour chaque canal ;
  - de réserver les ressources (RSPEC) pour chaque canal via l'API RSVP.

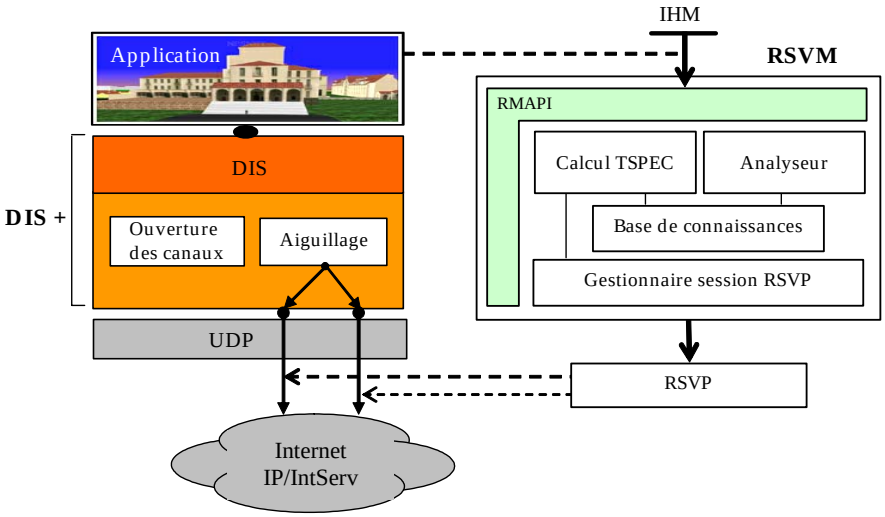


Figure 22 : Architecture de QdS de bout en bout DIS/ATM

1.1. DIS+

Le choix d'incorporer les fonctions de gestion des canaux dans DIS résulte du fait que les applications DIS ont une forte orientation militaire et utilisent des méthodes de cryptage des données avant soumission au système de communication (ici UDP/IP). De ce fait, l'identification des classes de PDU dans le trafic DIS est impossible à un niveau sous jacent.

En termes d'implantation, l'aiguillage des PDU se fait sur six canaux statiquement établis :

- un pour chacune des classes de PDU de gestion de la simulation, de PDU événementiels et de PDU de communication audio ;
- trois pour les PDU d'état (ES), concernant respectivement les entités terrestres, maritimes, et aériennes, dont les plages d'accélération maximales sont suffisamment distinctes.

1.2. RSVM

L'architecture du RSVM distingue cinq modules conceptuels :

- une API (RMAPI[151]) permettant à l'application ou un utilisateur averti :
  - d'ouvrir/fermer une session RSVM ;
  - d'ajouter ou de remplacer une entrée dans la base de connaissance décrite ci-après ;
  - de déclarer la morphologie et le nombre des entités gouvernées localement ;
  - de démarrer le processus de réservation de ressources ;
- une base de connaissances contenant une liste d'associations <morphologie, TSPEC> ;
- un analyseur de trafic chargé de mettre à jour dynamiquement la base de connaissance ;
- un module de calcul des paramètres du TSPEC : il prend en entrée la morphologie et le nombre d'entités indiqués via l'API, et s'appuie sur la base de connaissance ;
- un gestionnaire des sessions RSVP à établir pour chaque canal, chargé :
  - en émission, d'invoquer l'API RSVP pour initier l'établissement de la réservation[152] ;
  - en réception, d'effectuer, pour chaque message PATH reçu, la requête de réservation correspondante[153] via l'API RSVP.

2. Déploiement et mesures de performance

L'architecture DIS/ATM a été implémentée en C++ et testée pour les applications du projet sur une plateforme grande distance.

Cette plateforme, conçue dans le cadre du projet, interconnectait Toulouse (LAAS), Paris (LIP6) et Sophia Antipolis (INRIA), et s'appuyait sur un *backbone* ATM (SAFIR) fournissant un VP à débit constant entre Toulouse et Paris (resp. Sophia).

Des mesures de performance ont été effectuées sur la plateforme DIS/ATM pour évaluer la QdS obtenue pour des scénarios DIS reproductibles, sans, puis avec réservation de ressources. Les mêmes mesures ont également été effectuées en environnement Internet classique (donc sans réservation) entre Toulouse et Paris.

Ces mesures, détaillées dans [Garc02a], ont conduit aux conclusions suivantes :

- l'insuffisance du service Best Effort pour que les applications DIS puissent être distribuées correctement, a été démontrée au travers des résultats suivants :
  - sur DIS/ATM, la saturation progressive du VP Toulouse-Paris a provoqué une augmentation du délai de transit sans accroissement important du nombre de pertes ;

---

151 RMAPI : *Resource Manager Application Programming Interface*

152 Emission d'un message RSVP PATH.

153 Emission d'un message RSVP RESV

un contrôle de ressources s'est avéré nécessaire pour conserver un délai de transit compatible avec les besoins DIS, quelle que soit la charge du réseau ;

- sur Internet, et malgré l'incertitude quant au degré de charge du réseau, nous avons constaté une faible variation du délai de transit moyen, mais une plus grande variabilité du taux de pertes. Le service observé ne satisfaisait donc pas les exigences de fiabilité des applications DIS, et justifiait un besoin en contrôle de ressources[154] ;

– les mesures avec réservation de ressources n'ont pu être menées de façon concluante à cause des mécanismes de *policing* mis en œuvre à l'entrée de la plateforme SAFIR (au niveau ATM), non maîtrisés dans le contexte expérimental du projet. Cet aspect a été pris en compte lors du déploiement de l'architecture du projet @IRS.

V. Conclusion

Les travaux présentés dans cette étape visaient à proposer une architecture de bout en bout offrant des garanties de QdS à une famille d'applications complexes et temporellement contraintes (DIS), dans un environnement Internet de type IntServ.

Ces travaux nous ont conduit :

– à caractériser les besoins DIS de manière plus réaliste que le standard, notamment avec la prise en compte du délai de transit ;

– à proposer une classification plus fine des PDU DIS en vue d'optimiser l'utilisation des ressources réseau ;

– à définir comment traduire les paramètres applicatifs en paramètres réseau (TSPEC) ; notons ici que le point délicat est de caractériser le trafic pour la réservation, en particulier les pics de débit, dans des types d'applications ou les échanges ne constituent pas des flux réguliers (jeux distribués, flux vidéo avec images de taille variable, …).

– à définir et implémenter une architecture (DIS/ATM) masquant les réservations et l'aiguillage des données sur des canaux à QdS spécifiques.

Néanmoins, l'impact du Transport n'a pas été pris en compte ; il l'est dans la suite de nos travaux dans un contexte DiffServ, présentés ci-après dans l'Etape 3.

* * *

[154] Cette conclusion est renforcée par le fait que les performances observées entre Toulouse et Paris étaient vraisemblablement bien au-delà des performances nominales de l'Internet de l'époque.

Plan de l'Etape 3

I. Introduction 97

II. Architecture @IRS 97

1. Choix d'architecture 98

1.1. Principes généraux 98

1.2. Fonctions des plans de contrôle et de données du niveau IP 99

2. Implémentation et validation 99

2.1. Plateforme @IRS 100

2.2. Mesures de performance 100

III. Architecture @IRS++ 102

1. Extension des expérimentations @IRS par simulation (ns-2) 102

1.1. Topologie étudiées en simulation 103

1.2. Résultats et conclusions 104

2. Le modèle de performance 104

3. API et algorithme de sélection 105

3.1. API 105

3.2. Algorithme de sélection 106

IV. Architecture @IRS++/MD 108

1. Modèle de performance d'un canal en contexte multi domaines 109

1.1. Modèle mathématique sous-jacent à l'étude 110

1.2. Application à l'étude 110

2. Architecture 111

2.1. Approche adoptée pour @IRS++/MD 111

2.2. Signalisation 112

V. Conclusion 117

ETAPE 3 : CONTRIBUTION AUX ARCHITECTURES A QDS EN CONTEXTE DIFFSERV

I. Introduction

Partant des travaux précédents, nous avons poursuivi notre contribution aux architectures pour la QdS dans un nouvel environnement, DiffServ, en intégrant le Transport et en élargissant le domaine des applications candidates. Notre objectif était :

- d'offrir des garanties de QdS en termes d'ordre/fiabilité partiels et de délai, compatibles avec les contraintes minimales exprimées par l'application pour chacun de ses flux ;
- de minimiser l'utilisation des ressources réseau les plus coûteuses ;
- d'abstraire l'application du choix des services Transport et IP satisfaisant la QdS requise pour chacun de ses flux.

Les trois problématiques que nous avons successivement adressées ont été :

- d'abord, de définir l'architecture permettant d'obtenir une QdS IP par flux dans un environnement DiffServ mono domaine ; puis de valider ces choix par la caractérisation de la QdS obtenue ; ce travail a été réalisé dans le cadre du projet @IRS ;
- puis, de reconsidérer l'architecture @IRS, afin d'une part, d'abstraire l'application du choix des services Transport et IP sous jacents, et d'autre part d'optimiser l'utilisation des ressources ; ce travail a été réalisé dans le cadre du projet @IRS++ ;
- enfin, d'étendre l'architecture @IRS++ pour le passage au multi domaines ; ceci imposait de considérer les problèmes de provisionnement inter domaines[155] et de disponibilité des ressources[156], ce qui nous a en particulier conduit à aborder les besoins en signalisation correspondants. Ces travaux ont constitué nos apports initiaux au projet EuQoS.

La suite de ce chapitre est structurée comme suit : la section II expose les solutions apportées à la première problématique (architecture @IRS) ; la section III présente celles proposées face à la deuxième problématique (architecture @IRS++) ; nos propositions de réponse à la troisième problématique sont exposées en section IV (architecture @IRS++/MD).

II. Architecture @IRS

L'architecture @IRS a été conçue dans le cadre du projet RNRT du même nom, en particulier en collaboration avec le LIP6.

La problématique que nous avons adressée était double :

- définir une architecture permettant d'obtenir une QdS IP par flux dans un contexte DiffServ mono domaine, qui, par conception, manipule des classes en cœur de réseau ;
- valider ces choix par la caractérisation de la QdS obtenue.

[155] Choix de concaténation des classes de service le long du chemin de donnée.

[156] Sur les liens inter domaines.

Nous présentons d'abord l'architecture proposée, puis les principaux résultats des mesures de performance réalisées sur la plateforme nationale DiffServ constituée en conséquence.

1. Choix d'architecture

1.1. Principes généraux

Comme dans nos choix d'architecture effectués dans le contexte IntServ, le principe de base qui sous-tend l'architecture @IRS est que le trafic applicatif peut être décomposé en plusieurs flux, requérant chacun une QdS spécifique exprimable en termes de délai, de fiabilité, etc.

Au travers d'une *session*, l'utilisateur peut établir un ou plusieurs *canaux de communication de bout en bout*, chacun étant : (1) unicast, (2) dédié au transfert d'un seul flux, et (3) offrant une QdS spécifique au flux véhiculé (Figure 23).

Pour cela, il dispose d'une API[157] spécifique offrant les paramètres et les primitives de service nécessaires. Cet utilisateur peut être l'application elle-même, une IHM ou un module d'adaptation de l'application à l'API, similaire au RSVM proposé pour DIS dans l'Etape 2.

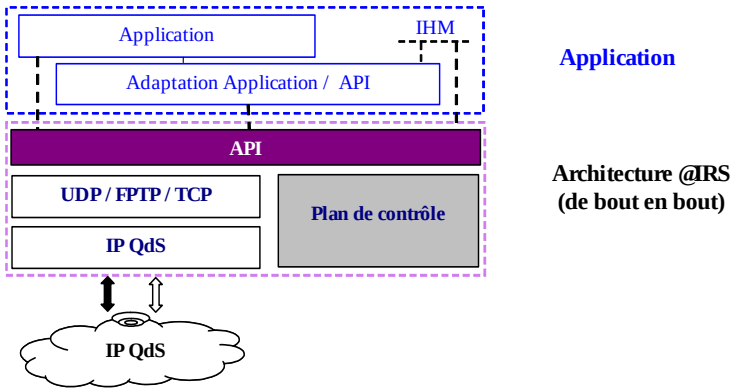


Figure 23 : Architecture @IRS de bout en bout

Outre l'API, l'architecture au niveau des hôtes comporte trois modules conceptuels :

- le premier permet l'accès à plusieurs services de Transport, respectivement fournis par TCP, UDP et un protocole à fiabilité partielle conçu sur les bases de FPTP [Expo03] ;
- le second permet l'accès à trois classes de service IP :
  - une classe analogue à la classe EF [Jaco99], visant un service garanti (GS) offrant un délai minimal et une fiabilité totale vis à vis des pertes par congestion des routeurs ;
  - une classe analogue à la classe AF [Hein99], visant un service assuré (AS), offrant une fiabilité quasi totale et un délai de transit aux variations « acceptables » ;
  - une classe par défaut visant le service Best Effort classique ;
- le troisième module identifie les fonctions et les protocoles du plan de contrôle pour établir les connexions de bout en bout, effectuer le contrôle d'admission, etc., par le biais d'une certaine signalisation.

[157] Nous décrirons cette API dans la partie relative à l'architecture @IRS++.

1.2. Fonctions des plans de contrôle et de données du niveau IP

Le modèle DiffServ suppose un provisionnement correct des domaines au regard des SLS souscrits. En contexte mono domaine, ce provisionnement peut être fourni par un sur dimensionnement des ressources[158], ou par des techniques d'ingénierie de trafic.

Pour @IRS (qui adresse le mono domaine), nous nous sommes placés sous l'hypothèse d'un sur provisionnement.

Cependant, DiffServ ne garantit pas la préservation de l'intégrité des flux au sein d'un agrégat. Il laisse néanmoins la possibilité de protéger les flux en bordure du réseau : c'est ce choix que nous avons fait, en effectuant dans les routeurs de bordure :

- au niveau du plan de contrôle :
  - un contrôle d'admission par flux, fonction de l'état courant de réservation de la bande passante allouée à la classe de service considérée ;
- au niveau du plan de donnée :
  - une classification par flux, incluant un *policing* conduisant à :
    - un écartement des paquets GS non conformes au profil de trafic annoncé ;
    - un marquage prioritaire à la perte (OUT) des paquets AS non conformes, et une politique spécifique de gestion de file d'attente[159] en cas de congestion [Bona00].

Enfin, et pour tous les routeurs du chemin de donnée (bordure et cœur), l'ordonnancement des paquets a été défini de la façon suivante :

- traitement prioritaire des paquets EF (*priority queuing*) ;
- traitement à priorité pondérée entre paquets AF et BE (*weighted fair queuing*).

Les schémas d'implantation de ces fonctions au niveau des routeurs sont illustrées Figure 24.

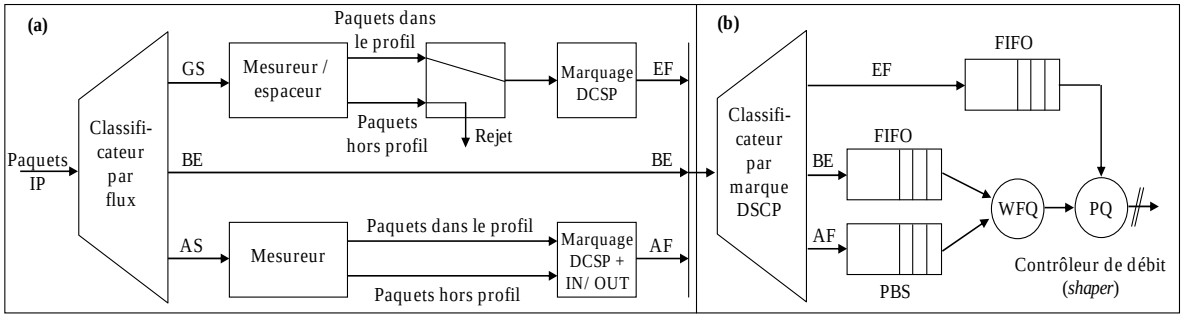


Figure 24 : Interfaces **(a)** d'entrée des routeurs de bordure **(b)** de sortie de tous les routeurs

2. Implémentation et validation

En collaboration avec le LIP6, nous avons implanté cette architecture sur une plate forme nationale, puis effectué des mesures de performance pour en valider les choix. Ce paragraphe résume les principaux résultats de ces travaux.

---

158 Ce qui n'est pas possible en contexte multi domaines.

159 *Partial Buffer Sharing* (PBS)

2.1. Plateforme @IRS

La plateforme @IRS (ou *@IRSBone*) a été construite sur les bases d'une plateforme ATM (RENATER2), offrant des VP de bout en bout de type CBR[160].

Quatre des sept partenaires étaient directement connectés au @IRSBone par le biais d'un routeur de bordure (R_b). Des routeurs de cœur (R_c) ont été introduits dans le @IRSBone ; physiquement, ils étaient situés sur les plates-formes des sites directement connectés, mais logiquement, ils appartenaient au @IRSBone : tout trafic entrant ou sortant d'un site passait obligatoirement par ces routeurs.

La Figure 25 illustre la connexion au @IRSBone des plateformes du LIP6 et du LAAS (les autres sites ne sont pas représentés).

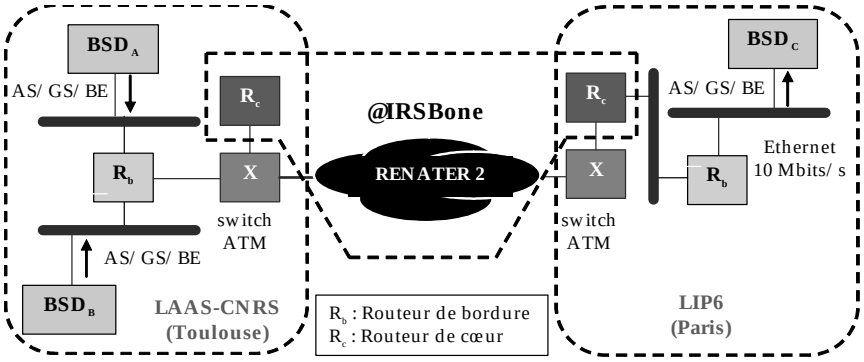


Figure 25. Connexion au @IRBone des plateformes locales du LIP6 et du LAAS

2.2. Mesures de performance

2.2.1. Configuration des éléments du @IRSBone

Les mesures de performance pour valider l'architecture ont été réalisées entre Toulouse et Paris sur la base des choix de configuration suivant :

- la bande passante du VP[161] ATM entre les deux sites était de 1 Mbits/s ;
- les routeurs de cœur et de bordure ont été configurés sous les hypothèses suivantes :
  - la quantité maximale QM_{GS} de trafic GS que pouvait introduire un routeur de bordure a été fixée à 20% de la bande passante du VP ATM ;
  - la quantité maximale QM_{AS} de trafic AS que pouvait introduire un routeur de bordure a été fixée à 40% de la bande passante du VP ATM ;
  - les poids du WFQ appliqué entre paquets AF et BE étaient de 0,5 et 0,5.

2.2.2. Spécification des mesures et principaux résultats

Deux campagnes de mesures de performance ont été successivement menées ; nous ne détaillerons que la seconde, dont les résultats ont une conséquence directe sur le modèle de performance des services IP défini pour l'architecture @IRS++.

---

[160] CBR : *Constant Bit Rate*

[161] VP *: Virtual Path*

La première campagne visait à évaluer la QdS obtenue pour :

- un flux GS (respectivement AS) généré seul à différents niveaux de charge, et confronté à une charge progressive du réseau en trafic BE, jusqu'à saturation ;
- un flux GS et un flux AS générés conjointement à différents niveaux de charge, et confrontés à une charge progressive du réseau en trafic BE, jusqu'à saturation.

Les résultats ont permis de conclure positivement sur l'adéquation des choix d'architecture aux caractéristiques des services attendues : protection parfaite du flux GS et accroissement acceptable du délai pour le flux AS. Ces résultats sont détaillés dans [Garc01].

La deuxième campagne s'appuyait sur trois scénarios ayant pour but d'évaluer, dans le cas d'un réseau saturé en trafic BE, l'impact du nombre et de la charge des flux :

- AS sur la QdS AS, en l'absence de flux GS (scénario 1) ;
- GS sur la QdS GS, en l'absence de flux AS (scénario 2) ;
- AS et GS conjoints, sur les QdS AS et GS (scénario 3).

Seuls les deux premiers sont présentés ci-après. Les résultats complets de ces mesures sont disponibles dans [Chas02] [Garc02b].

Dans le premier scénario, plusieurs flux AS sont émis depuis les PC BSD_A et BSD_B (Toulouse) à destination du PC BSD_C (Paris). Trois cas de figure sont considérés : dans chacun, la somme des flux AS représente la moitié de la quantité autorisée QM_{AS} pour la classe de trafic AS ; la répartition entre les flux individuels est donnée dans le Tableau 3. Parallèlement, 1 ou 2 flux BE (selon les cas) sont générés, saturant la bande passante du VP ATM.

Le deuxième scénario est analogue au premier en remplaçant GS par AS.

Scénario 1	% de QM_{AS}	Débit du trafic BE (% de la BP du VP ATM)
AS (BSD_A)	50	100 (BSD_B)
AS1 (BSD_A) AS2 (BSD_B)	23 27	100 (50% BSD_A - 50% BSD_B)
AS11 (BSD_A) AS12 (BSD_A) AS21 (BSD_B) AS22 (BSD_B)	11 14 13 12	100 (50% BSD_A - 50% BSD_B)

Tableau 3 : Spécification du trafic pour le scénario 1
(Remplacer AS par GS pour le scénario 2)

Les résultats obtenus pour les deux scénarios sont présentés Figure 26.

Ils permettent de conclure que, pour des flux respectant leur profil de trafic, l'impact du nombre et de la charge des flux :

- GS sur la QdS GS est faible, la différence de délai provenant de la mise en attente d'au plus un paquet dans la file ;
- AS sur la QdS AS est faible pour 90% des paquets, mais 10% subissent un délai nettement accru. Ce résultat est acceptable au regard de la spécification du service AS.

Par ailleurs, les résultats du scénario 3 ont permis de conclure que l'impact de la charge d'un flux AS (resp. GS) sur la QdS GS (resp. AS) était quasiment nul.

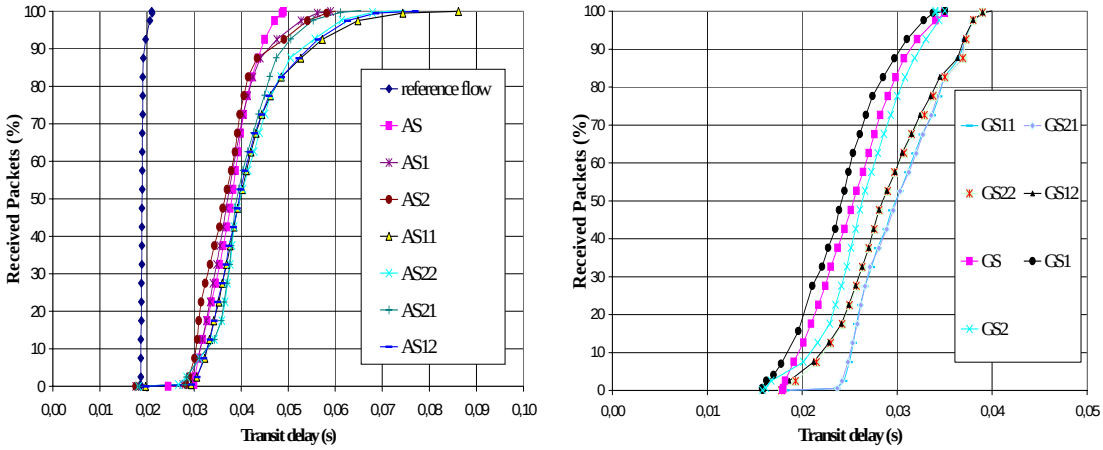


Figure 26 : Résultats du scenario 1 (gauche) / scenario 2 (droite)

III. Architecture @IRS++

L'architecture @IRS++ a été conçue dans le cadre du projet RNRT du même nom, qui faisait suite au projet @IRS. La problématique que nous avons adressée était :

- d'abstraire le niveau applicatif du choix des services sous jacents, en lui permettant d'exprimer ses besoins en QdS, indépendamment des services IP et Transport ;
- de répondre à ces besoins en optimisant l'utilisation des ressources du réseau, par le choix du meilleur couple {Transport / IP} satisfaisant les contraintes de l'application ; trois services ont été considérés au niveau IP : GS, AS et BE. Trois protocoles ont été considérés au niveau Transport : TCP, UDP et un protocole à fiabilité partielle, $FPTP_n$, supposé paramétrable en nombre maximum n de retransmissions.

Nous avons répondu au premier problème en définissant une API, incluant des paramètres de QdS génériques ainsi que des sémantiques de garantie pour ces paramètres.

Pour répondre au deuxième problème, le besoin était de définir :

- un modèle des performances d'un canal de bout en bout (couplant Transport et IP) ;
- un algorithme, qui sur ces bases, sélectionnait le couple {Transport/classe IP} satisfaisant les contraintes minimales de l'application, à moindre coût en termes de ressources réseau.

Concernant le premier besoin, les mesures effectuées sur le @IRSBone esquissaient un modèle de performance d'un canal IP basé sur une fonction de répartition du délai de transit. Sur ces bases, nous avons cherché à conforter ce modèle par une extension des cas envisagés dans les mesures ; les limites du @IRSBone nous ont conduit à une étude en simulation (ns-2) sur une topologie de réseau plus complexe.

L'étude menée en simulation est résumée au paragraphe (1) ci-après. Le modèle des performances d'un canal de bout en bout est ensuite présenté au paragraphe (2). L'API et l'algorithme de sélection sont enfin décrits au paragraphe (3).

1. Extension des expérimentations @IRS par simulation (ns-2)

Cette étude avait pour but de conforter les résultats obtenus pour les flux AS. Du fait du type d'ordonnancement appliqué entre flux AS et BE, la caractérisation du service AS semblait en effet problématique lorsque le nombre de sites et de routeurs augmentait.

La topologie de la plateforme @IRS (cf. Figure 25) se limitant à deux émetteurs (BSD_A et BSD_B) et un récepteur (BSD_C), la poursuite de l'étude visait à considérer une topologie plus complexe, et ne pouvait être menée qu'en simulation.

1.1. Topologie étudiées en simulation

La simulation étend la topologie vis à vis :

– du nombre d'émetteurs par site local (BE et ASi de la Figure 27) ;

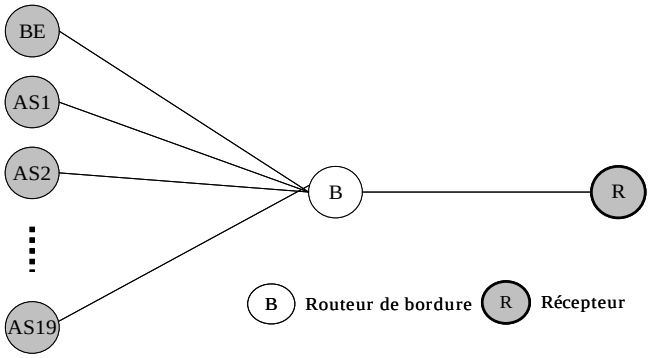


Figure 27 : Augmentation du nombre de sources de trafic dans un site local

– du nombre de routeurs de cœur, chargé chacun par un site local (Ci sur la Figure 28) ;

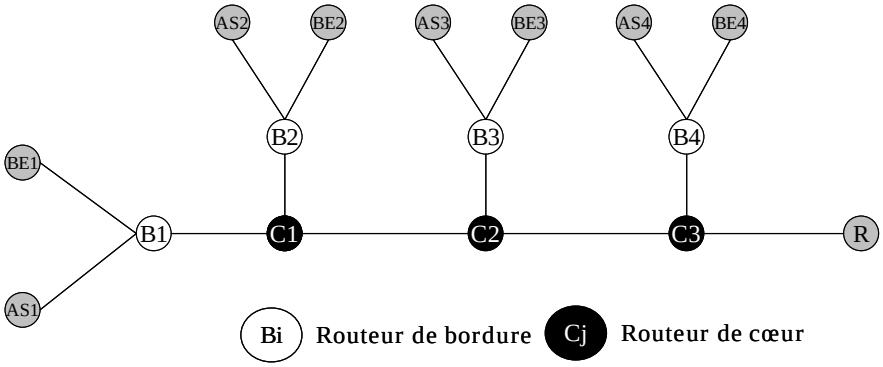


Figure 28 : Augmentation du nombre de routeurs de cœur

– du nombre de sites locaux convergeant sur un routeur de cœur (Bi sur la Figure 29).

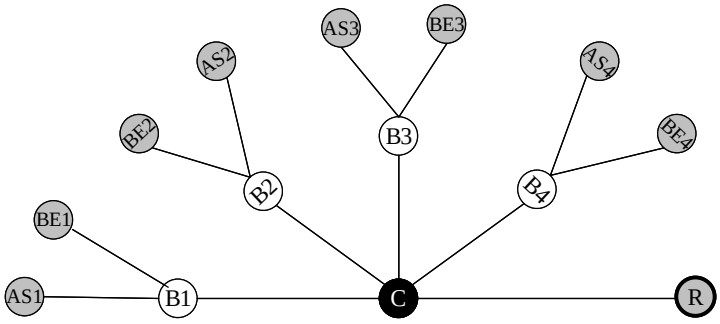


Figure 29 : Augmentation du nombre de sites locaux sur un routeur de cœur

1.2. Résultats et conclusions

Les résultats de simulation sont analogues à ceux obtenus sur la plateforme @IRS pour les deux premiers cas.

Pour le troisième cas 3, l'augmentation du nombre de sites locaux convergeant sur un routeur de cœur fait apparaître un accroissement du délai au delà d'un certain pourcentage de paquets ; ce pourcentage s'abaisse d'autant que le nombre de sites augmentent.

Ce phénomène apparaissait déjà dans le cas des flux AS_{11} AS_{12} AS_{21} AS_{22} du scénario 1 (Figure 26), assimilable à deux sites locaux convergeant sur un même routeur.

Cependant, l'accroissement du délai consécutif à ce phénomène reste limité du fait du nombre restreint d'interfaces physiques que présente en général un routeur de cœur.

Ces résultats de simulation, couplés aux mesures, nous ont conduit à poser l'hypothèse que *les performances obtenues au niveau IP pour un flux à QdS sont indépendantes de la quantité*[162] *de flux à QdS circulant dans le réseau et de la topologie de celui-ci, pour une charge en trafic BE donnée et une route donnée.*

Dans notre étude, ces performances ont été obtenues dans le pire cas d'un trafic BE saturant les liens. Les détails de cette étude sont fournis dans [Auri03].

2. Le modèle de performance

Le modèle de performance d'un canal de bout en bout part de l'hypothèse précédente, qui caractérise les performances d'un canal IP à QdS par la fonction de répartition du délai de transit de chaque paquet.

Cependant, la QdS perçue par l'application résulte du couplage entre le service IP et les mécanismes de Transport ayant un impact sur le délai, typiquement la reprise des pertes.

Le modèle de performance vise donc à calculer la fonction de répartition du délai pour les PDU de niveau Transport, à partir de la fonction de répartition de niveau IP et de la prise en compte des retransmissions au niveau Transport.

Soient :

- $f(t)$ la fonction de répartition du délai de transit des paquets IP entre 2 sites ;
- $f_{n,T}(t)$ la fonction de répartition du délai des PDU Transport après n retransmissions, T désignant la valeur du *timer* de retransmission des PDU Transport (supposée constante) ;
- ε le taux de perte moyen observé sur le flux : $\varepsilon = \lim_{t\to\infty}(1 - f(t))$;

Alors :

$$f_{1,T}(t) = f(t) + \varepsilon \,.\, f(t - T)$$

La fonction $f_{n,T}(t)$ de répartition du délai de transit après n retransmissions peut donc être exprimée par :

$$f_{n,T}(t) = \sum_{i=0}^{n} \varepsilon^i \,.\, f(t - i.T)$$

[162] Nombre et charge

3. API et algorithme de sélection

L'algorithme de sélection automatique des services Transport et IP s'appuie sur le modèle de performance précédent pour satisfaire (si possible) les paramètres de QdS fournis par l'utilisateur lors de l'invocation du service @IRS++, via son API.

Nous présentons tout d'abord cette API, puis l'algorithme de sélection.

3.1. API

L'API @IRS ++ permet d'accéder au service @IRS++ de manière transparente, c'est à dire sans avoir conscience des services Transport et IP sous jacents. Elle comporte plusieurs primitives de service permettant de réclamer :

- l'établissement/fermeture d'une session ;
- l'établissement/fermeture d'un canal de bout en bout à QdS, dans le cadre d'une session ;
- l'émission ou la réception d'un message sur un canal donné.

L'expression de la QdS requise se fait au moyen de paramètres de QdS et de sémantiques de garantie souhaitées pour ces paramètres.

3.1.1. Paramètres de QdS

Les paramètres de QdS que l'application doit expliciter via l'API @IRS++ sont :

- une fiabilité partielle et un délai de transit, exprimés en termes de :
  - % de paquets devant être reçus : τ_r ;
  - % de paquets devant être reçus dans l'intervalle de temps [a,b] : τ_d ([*a*,*b*]) ;
  - nombre maximal de pertes consécutives admissibles ;
- un ordre partiel inter flux, permettant à l'application d'exprimer des contraintes de synchronisation logiques entre des flux différents ;
- une caractérisation du débit exprimée sous la forme d'un *token bucket*.

3.1.2. Sémantiques de garantie

Trois sémantiques de la garantie offerte par le système de communication sur les paramètres τ_r et (τ_d, [*a*,*b*]) ont été définies.

Elle sont respectivement notées σ_r et σ_d et peuvent prendre trois valeurs :

- la garantie absolue, notée *A* ;
  - elle signifie que la valeur du paramètre doit être obtenue de façon sûre, par un service garanti au niveau IP, ou par un mécanisme de retransmissions au niveau Transport ;
  - la requête de cette garantie sur le paramètre τ_d implique le respect du profil de trafic annoncé ; en cas de violation, la garantie n'est pas assurée ;
  - DIS est un exemple d'applications requérant ce type garantie. Un flux de données événementielles (tir, détonation, etc.) doit par exemple être servi selon cette garantie tant sur τ_r que sur τ_d, car une altération de la fiabilité ou du délai de transit de ces données est préjudiciable à la cohérence de la simulation.

– la garantie en moyenne, notée *M* (ou *N*) ;
  - elle signifie que la valeur du paramètre peut être obtenue, non plus de façon certaine, mais statistiquement. Une incertitude apparaît donc dans les performances du service fourni à l'application ;
  - les applications de visioconférence peuvent réclamer une telle sémantique tant sur τ_r que sur τ_d (au moins pour le flux vidéo), car les utilisateurs finaux sont susceptibles de tolérer une altération de la vidéo ;

Notons que cette garantie peut être déclinée de deux façons suivant qu'elle notifie (N) ou non (M) la dégradation de qualité à l'application utilisatrice. Les applications adaptatives tirent partie de cette possibilité.

3.2. Algorithme de sélection

L'algorithme de sélection automatique des services repose sur l'hypothèse que les paquets d'un même flux sont indépendants en probabilité vis à vis de la perte et du délai. En effet, les congestions des routeurs entraînent des rafales de pertes portant sur des paquets qui, le plus souvent, n'appartiennent pas à un même flux.

Nous présentons les principes de cet algorithme. Les détails de l'étude correspondante sont fournis dans [Chas03].

3.2.1. Principes de l'algorithme

Afin de juger si une requête applicative peut être satisfaite, il s'agit successivement :
– de vérifier la cohérence de la requête et de rejeter celle-ci en cas d'incohérence ;
– puis en cas de cohérence :
  - d'effectuer le choix du couple (Transport/IP) permettant de satisfaire la requête ;
  - d'effectuer ensuite un contrôle d'admission, dès lors que le service IP retenu diffère du *Best Effort* ; en cas d'échec, il s'agit alors de revenir au point précédent et de tester un autre couple s'il y en a. En cas d'échec pour tous les couples, la requête est rejetée.

Trois services ont été considérés au niveau IP : GS, AS et BE. Trois protocoles ont été considérés au niveau Transport : TCP, UDP et $FPTP_n$, n désignant le nombre maximum de retransmissions autorisées.

Vérification de la cohérence de la requête

Pour qu'une requête soit jugée cohérente, deux conditions évidentes doivent être vérifiées :
– le taux des paquets que souhaite recevoir l'application doit être supérieur au taux de paquets que souhaite recevoir l'application dans les délais, soit : $0 \leq \tau_d \leq \tau_r \leq 1$;
– le délai de transit maximal, b, requis par l'application, doit être supérieur au délai de transit minimum du réseau, noté t_0, observé entre les sites considérés, soit : $b \geq t_0$.

Sélection du couple (Transport / IP)

Le choix du meilleur couple (Transport / IP) repose sur les fonctions :
– f_{GS} caractérisant les performances du canal de bout en bout pour GS ;
– $f_{AS,n}$ caractérisant les performances du canal pour AS après n retransmissions.

Pour AS, il s'agit donc de déterminer le nombre n de retransmissions des PDU Transport permettant de satisfaire la fiabilité requise τ_r tout en respectant le taux de paquets τ_d reçus avant le délai maximal b.

Soit :

- n_r le nombre de retransmissions nécessaires à la satisfaction du paramètre τ_r
  - le taux de fiabilité obtenu après x retransmissions étant $1-\varepsilon^{x+1}$, il en résulte que :

$$\tau_r \leq 1-\varepsilon^{n_r+1} \Rightarrow n_r \geq \text{Ent}\,[\ln(1-\tau_r)/\ln(\varepsilon)]$$

    - où : ln désigne le logarithme népérien et Ent la partie entière ;
    - Note : si $\varepsilon = 0$ alors $n_r = 0$;
- n_d le nombre maximum de retransmissions compatible avec le respect du délai maximal requis b :
- $n_d = \text{Ent}\,[(b-t_0)/T]$

Le nombre n est le plus petit entier vérifiant : $n_r \leq n \leq n_d$ et $f_{AS,n}(b) \geq \tau_r$.

Le choix est basé sur le Tableau 4, qui indique, pour toutes les configurations de (τ_r, σ_r) et (τ_d, σ_d), un ou plusieurs couples (Transport, IP) permettant de satisfaire la requête. Lorsque plusieurs couples sont envisageables, ils sont classés par ordre de coût[163] et le couple à coût minimal est sélectionné.

<table>
<tr><td>τ_d \ τ_r</td><td>$\tau_r = 0$</td><td colspan="3">$0 < \tau_r < 1$</td><td>$\tau_r = 1$</td></tr>
<tr><td>$\tau_d = 0$</td><td>UDP/BE
∀ la sémantique</td><td colspan="3">FPTP/BE
∀ la sémantique A, M ou N</td><td>TCP/BE
∀ la sémantique</td></tr>
<tr><td rowspan="3">$0 < \tau_d < 1$; [a,b]</td><td rowspan="3"></td><td>σ_d \ σ_r</td><td>A</td><td>M/N</td><td rowspan="3">UDP/GS
∀ la sémantique</td></tr>
<tr><td>A</td><td>UDP /GS</td><td>UDP /GS</td></tr>
<tr><td>M/N</td><td>UDP /GS</td><td>Choix</td></tr>
<tr><td>$\tau_d = 1$; [a,b]</td><td></td><td colspan="3"></td><td>UDP/GS
∀ la sémantique</td></tr>
</table>

Tableau 4 : Correspondance entre les paramètres de QdS et les services IP et Transport (les hachures indiquent une incohérence dans la requête)

La première ligne du tableau ($\tau_d = 0$) correspond à une requête de QdS sans contrainte temporelle. On retrouve donc un choix basé sur le Transport uniquement pour un service IP toujours *best effort*.

La dernière colonne ($\tau_r = 1$) correspond à une requête de QdS à fiabilité totale. En présence de contrainte temporelle, celle ci ne peut être garantie qu'avec un service GS au niveau IP. De ce fait, la reprise des pertes n'a pas lieu d'être et le choix se porte sur UDP au niveau Transport.

Le bloc en grisé clair restant correspond à une fiabilité partielle ($0 \leq \tau_r \leq 1$) et à une contrainte temporelle ($0 \leq \tau_d \leq 1$, pour un délai inférieur à b) :

- si la sémantique de garantie requise pour l'un ou l'autre des paramètres est la garantie absolue (A), alors seul le service GS au niveau IP est susceptible d'assurer cette garantie ; de ce fait, comme précédemment, le choix se porte sur UDP au niveau Transport ;

[163] Coût du service GS > coût du service AS > coût du service BE

– si la sémantique de garantie pour les deux paramètres est statistique (M ou N), il convient alors d'envisager successivement les couples UDP/AS, FPTP$_n$/AS ou UDP/GS. Le choix est effectué comme suit :
  - on considère d'abord la fonction de répartition pour AS
    – si $\tau_d \leq f_{AS,0}(b)$ et si $\tau_r \leq 1 - \varepsilon$ alors le couple UDP/AS convient ;
    – sinon, on envisage alors FPTP$_n$ pour lequel on détermine la valeur de n selon l'algorithme suivant :

```
success = false
n = n_r
while ((n ≤ n_r) and not success)
  if (f_AS,0(b) ≥ τ_d) then success = true
  else n = n+1
```

    en cas de succès, le couple FPTP$_n$/AS convient ;
  - en cas d'insuccès avec AS, on considère la fonction de répartition pour GS
    – si $\tau_d \leq f_{GS}(b)$ alors le couple UDP/GS convient.

3.2.2. Exemples d'application

Nous illustrons ici la mise en œuvre de l'algorithme pour deux exemples simples :

<u>*Exemple 1*</u>. Pour une application requérant : $\tau_r = 0.75$, $\sigma_r = M$ et $\tau_d \leq 0.70$, $b = 125$ *ms*, $\sigma_d = A$, il n'y a qu'un seul choix possible du fait de la sémantique absolue (A) requise pour τ_d : le couple (UDP, GS) ; ce couple permet de satisfaire la requête si $f_{GS}(b=125\ ms)$ est supérieur à *0.70* ;

<u>*Exemple 2*</u>. Pour une application requérant : $\tau_r = 0.80$, $\sigma_r = M$ et $\tau_d \leq 0.70$, $b = 125$ *ms*, $\sigma_d = M$, la case « Choix » du Tableau 4 comporte plusieurs solutions. L'algorithme sélectionnera par ordre de préférence décroissant :

– le couple (UDP, AS) si $\tau_r \leq 1 - \varepsilon$ et $\tau_d \leq f_{AS,0}(b)$
– le couple (FPTP$_n$, AS) si $\tau_r \leq 1 - \varepsilon^n$ et $\tau_d \leq f_{AS,n}(b)$
– le couple (UDP, GS) si $f_{GS}(b=125\ ms)$ est supérieur à *0.70*.

L'architecture @IRS++ a été conçue dans un contexte mono domaine. Ce n'est que dans un second temps que nous avons considéré le contexte du multi domaines. La section suivante décrit la partie de nos travaux visant à définir comment étendre l'architecture @IRS++ pour offrir le même service (via la même API) dans ce nouveau contexte.

IV. Architecture @IRS++/MD

Le passage au multi domaines nous a conduit à reconsidérer la proposition @IRS++ vis à vis de deux problèmes importants : le provisionnement de services et la disponibilité des ressources sur le chemin de donnée.

En effet, dans l'architecture @IRS++ (conçue pour le mono domaine) :

– la réponse au premier problème suppose que les performances des services offerts par le domaine[164] sont connues ou accessibles par l'hôte émetteur au moment de la requête de création d'un canal à QdS[165] ;

[164] Fonction de répartition du délai de transit

- pour le multi domaines, la disponibilité des performances des services de bout en bout est moins évidente à assurer car les données peuvent traverser plusieurs domaines non connus à l'avance, mais fonction d'une destination précisée au moment de la requête. En conséquence, le domaine d'accès ne dispose pas a priori de cette information, comme cela était envisageable pour le cas du mono domaine ;

– la réponse au deuxième problème repose sur un contrôle d'admission portant uniquement sur le routeur d'entrée du domaine ;
  - pour le multi domaines, cette solution ne suffit plus car le sur provisionnement à l'échelle du multi domaines n'est pas réaliste : un contrôle d'admission sur chacun des domaines traversés et sur les liens inter domaines est nécessaire.

Répondre à ces limites nous a conduit à étendre le plan de contrôle de l'architecture @IRS++, et ainsi, à aborder le thème de la signalisation pour l'Internet multi domaines.

Notre démarche a suivie deux étapes :

– la première a été de reconsidérer le modèle de performance d'un canal de bout en bout à QdS, au regard du contexte multi domaines ciblé ;

– sur ces bases, nous avons proposé une solution couplée pour le provisionnement et le CAC inter domaines, visant, comme pour le mono domaine, à minimiser l'utilisation des ressources du réseau et à abstraire l'utilisateur du choix des services Transport et IP. Pour cette deuxième étape, nous avons spécifié et implémenté les protocoles de signalisation nécessaires, avant de les intégrer à l'architecture @IRS++.

Ces deux étapes sont successivement présentées aux paragraphes (1) et (2) suivants.

1. Modèle de performance d'un canal en contexte multi domaines

Dans le contexte du multi domaines, les données peuvent potentiellement traverser plusieurs domaines dont les performances entre routeurs d'entrée et de sortie du tronçon intra domaine emprunté sont différentes (Figure 30).

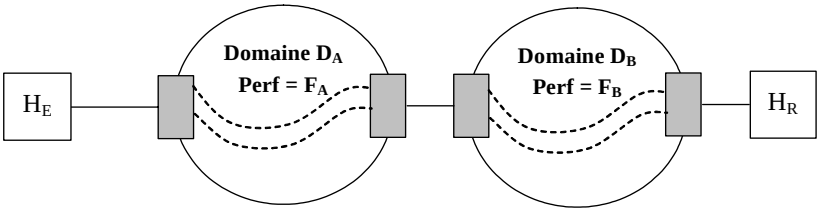


Figure 30 : Cas du multi domaines

Pour modéliser les performances du canal de bout en bout (Transport et IP), il s'agit donc, d'abord de caractériser les performances du canal IP résultant de la concaténation des canaux IP intra domaines traversés, et ensuite, de prendre en compte l'impact des retransmissions de niveau Transport.

Les deux paragraphes suivants décrivent notre réponse au premier besoin uniquement. En effet, elle nous conduit, comme pour le mono domaine, à caractériser le canal IP de bout en bout par la fonction de répartition du délai de transit ; la réponse au second besoin est alors identique au cas du mono domaine.

---

[165] C'est sur ces bases, et à partir des paramètres de QdS exprimées via l'API @IRS°++, que la requête de réservation pour la classe de service la moins coûteuse est effectuée (en tenant compte de l'impact des retransmissions du niveau Transport).

1.1. Modèle mathématique sous-jacent à l'étude

Nous rappelons ici les propriétés mathématiques nécessaires à notre étude.

Variable aléatoire

Dans un espace probabilisé (Ω, τ) - où Ω est l'ensemble des éventualités (univers) et τ une classe de parties de Ω - on associe à tout événement $e \in \tau$, le nombre réel $P(e)$ appelé probabilité de la réalisation de e ; on appelle *loi de probabilité,* l'application P de τ dans [0,1] telle que $P(\Omega)$=1et $P(\varnothing)$=1.

Une *variable aléatoire* X est une application d'un espace probabilisé (Ω, τ, P) dans $\mathbb{R}$.

Fonction de répartition et densité de probabilité d'un variable aléatoire

On définit alors la *fonction de répartition* F_X de la variable aléatoire X par :

$$F_X(u) = P\left(\left\{e \in \tau \middle| X(e) < u\right\}\right)$$

$F_X(x)$ est une fonction non-décroissante et continue à gauche ; sa dérivée[166] f_X constitue la *densité de probabilité* de la variable aléatoire X :

$$f_X(u) = \frac{d}{du} F_X(u) \text{ soit encore } F_X(u) = \int_{-\infty}^{u} f_X(t) \cdot dt$$

Produit de convolution de densités de probabilités

On appelle produit de convolution de deux fonctions f et g la fonction h notée $h=f * g$:

$$h(t) = \int_{-\infty}^{\infty} f(t-x) \cdot g(x) \cdot dx$$

Notons que le produit de convolution est commutatif et associatif.

Si X et Y sont deux variables aléatoires réelles indépendantes, de densité de probabilité f_X et f_Y, alors la densité de probabilité de la variable aléatoire $Z= X+Y$ est $f_Z=f_X * f_Y$.

Des propriétés de la convolution il résulte que la fonction de répartition de Z est la dérivée du produit de convolution des fonctions de répartition de X et de Y.

1.2. Application à l'étude

Soit un flux à QdS dont les paquets IP doivent traverser 2 domaines A et B.

Soient alors :

- X *(*resp. Y*)* le délai de transit d'un paquet dans le domaine A (resp. B)
- F_X (resp. F_Y) la fonction de répartition du délai de transit[167] dans le domaine A (resp. B).

X et Y peuvent être considérés comme deux variables aléatoires prenant leurs valeurs dans un intervalle $[t_{min}, \infty[$, où t_{min} représente le plus petit délai et ∞ le cas d'une perte.

---

[166] Eventuellement au sens des distributions pour une v.a. discrète ou mixte.

[167] Entre deux routeurs de bordure et pour un état de charge du domaine, par exemple un état de congestion.

D'après l'hypothèse d'indépendance en probabilité, la fonction de répartition du délai de bout en bout F_Z avec $Z=X+Y$ peut être définie par :

$$\boldsymbol{F_Z(t) = \frac{d}{dt}\left(F_X(t) * F_Y(t)\right)}$$

La généralisation à la traversée de n domaines se fait à partir du résultat obtenu avec $n-1$ domaines grâce à la propriété d'associativité du produit de convolution :

$$\boldsymbol{F_{Z_n}(t) = \frac{d}{dt}\left(F_{X_n}(t) * F_{Z_{n-1}}(t)\right)} \quad \forall\, n \geq 2 \text{ et } \boldsymbol{F_{Z_1}(t) = F_{X_1}(t)}$$

où : $\boldsymbol{F_{X_n}(t)}$ désigne la fonction de répartition du délai des paquets traversant le domaine n.

En conclusion, cette étude montre que le modèle de performance défini en contexte mono domaine s'applique aussi au cas du multi domaines. Ceci permet également d'étendre le champ d'application de l'algorithme de sélection automatique des services au contexte du multi domaines.

A partir de ces résultats, nous avons considéré le passage au multi domaines, en cherchant à définir comment étendre l'architecture @IRS++ pour offrir un service analogue à celui proposé dans le mono domaine.

La section suivante présente la proposition d'architecture faite en ce sens, sur les bases de l'architecture @IRS++.

2. Architecture

L'architecture @IRS++/MD propose des solutions aux deux problèmes importants que sont le provisionnement des services (pour l'aspect concaténation des classes de services[168]) et la disponibilité des ressources à l'échelle du multi domaines, déjà discutés dans l'état de l'art.

Cette section expose tout d'abord l'approche que nous avons adoptée, puis les travaux effectués en conséquence sur le thème de la signalisation.

2.1. Approche adoptée pour @IRS++/MD

Notre approche vis à vis du provisionnement du service, au sens choix concaténation des classes :

- adopte le point de vue utilisateur, qui cherche à évaluer les performances des services de bout en bout, avant d'invoquer le plus adéquat au regard des besoins applicatifs ;
- est indépendante du type de version de BGP, et du type de routage inter domaines appliqué en conséquence (orienté QdS ou non).

Vis à vis de la disponibilité des ressources, et en conséquence du besoin en CAC, notre approche s'inscrit dans le modèle à base de *Bandwidth Broker* (BB), qui, par principe, est plus générique que celui n'impliquant que les éléments du chemin de donnée.

Dans notre proposition d'architecture, le provisionnement et le CAC sont effectués dynamiquement suite à une requête de création d'un canal à QdS[169]. Ils sont mis en œuvre au sein du plan de contrôle.

---

[168] L'aspect dimensionnement des ressources intra domaine est hors de notre propos.

Le provisionnement se résout en trois étapes (Figure 31) :

- 1) découverte des classes de service et de leurs performances auprès de chacun des domaines du(des) chemin(s) inter domaines mis à disposition par BGP ;
- 2) évaluation par le biais de la convolution, du modèle de performance de l'ensemble des classes disponibles de bout en bout sur le(les) chemin(s) ;
- 3) choix d'une classe, qui couplée au Transport, satisfait la QdS requise ; ce choix repose sur l'algorithme de sélection automatique, appliqué à présent au modèle de performance des classes disponibles de bout en bout, et non plus seulement au modèle intra domaine.

Le CAC est couplé au provisionnement :

- durant l'étape 1 du provisionnement, une pré-réservation des ressources nécessaire est effectuée au niveau du BB pour chacune des classes disponibles sur le chemin de donnée ; chaque BB vérifie la conformité de la requête vis à vis du contrat souscrit en entrée, compte tenu de l'état courant d'utilisation de ce contrat[170] ;
- à l'issue du choix de l'étape 3, les pré-réservations pour la classe retenue sont confirmées, les autres sont relâchées.

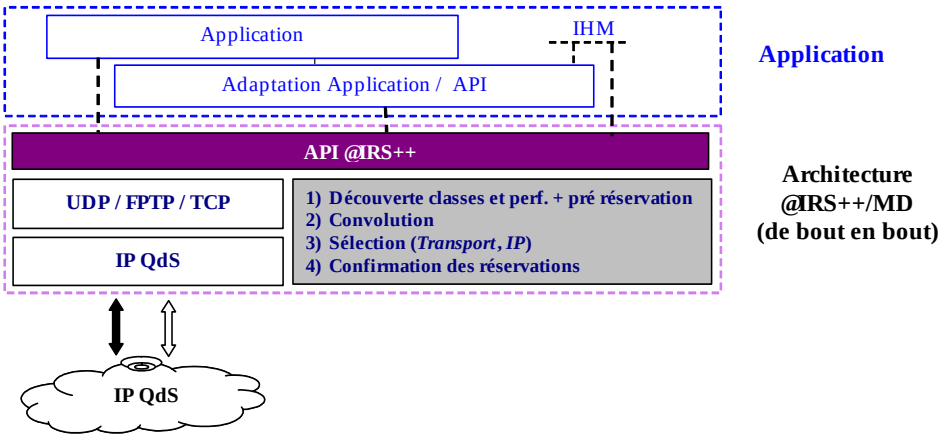


Figure 31 : Architecture @IRS++/MD de bout en bout

2.2. Signalisation

L'implantation du plan de contrôle de l'architecture @IRS++/MD (partie grisée de la Figure 31) nous a conduit à aborder le thème de la signalisation pour un modèle d'Internet multi domaines à base de BB.

L'état de l'art a présenté les travaux du groupe NSIS de l'IETF, menés sur le thème de la signalisation. NSIS distingue deux types de signalisation, *couplée* et *découplée du chemin de donnée*, et focalise actuellement son attention sur le premier type.

Notre proposition s'inscrit dans le second type, mais elle a été conçue indépendamment du cadre architectural à deux niveaux[171] introduit par NSIS. Son adaptation à ce cadre général fait partie de notre prospective pour une signalisation découplée du chemin de donnée.

---

[169] Exprimée en les paramètres de l'API @IRS++ : ordre partiel inter flux, fiabilité (τ_r), délai (τ_d, [a, b]), sémantique de garantie associées (A, M, N) et caractérisation du profil de trafic (*token bucket*).

[170] Chaque domaine est supposé bien provisionné en interne, ainsi l'appel à une fonction de CAC intra domaine n'apparaît pas.

[171] NTLP (Transport spécifique à la signalisation) et NSLP (signalisation proprement dite).

2.2.1. Hypothèses

Notre proposition s'appuie sur les hypothèses suivantes :

- homogénéité du modèle de QdS pour tous les domaines ;
- connaissance, au niveau de chaque BB, des performances[172] des classes de service IP, intra ou inter domaines, supportées pour tout couple de routeurs de bordure ;
- unicité des BB au sein chaque domaine.

2.2.2. Protocole

2.2.2.1. Equipements impliqués

Les équipements impliqués dans la signalisation (cf. Figure 32) sont :

- les hôtes émetteur et récepteur, supposés clients de leurs domaines d'accès respectifs ;
- le routeur de bordure par lequel l'hôte émetteur accède à son domaine de souscription ;
- les BB de chaque domaine.

Pour ces derniers, nous distinguons le BB du domaine d'accès de l'hôte émetteur (appelé *BB local*) du(des) BB du(des) domaine(s) suivant(s) sur le chemin (appelé(s) *BB distant(s)*).

L'hôte émetteur est supposé connaître l'adresse de son BB local.

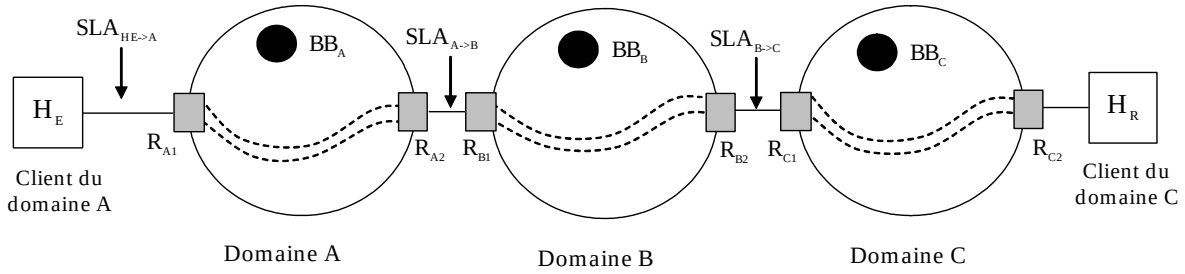


Figure 32 : Modèle d'Internet multi domaines

2.2.2.2. Principaux PDU et règles d'échange

Seuls sont ici décrits les principaux PDU et leurs règles d'échange entre les équipements impliqués. Les machines à états du protocole sont disponibles dans [Auri04a].

Quatre types d'échanges sont distingués :

- le type 1 concerne les échanges entre les hôtes émetteur et récepteur (H_E et H_R Figure 32) :
  - 2 PDU sont échangés pour la création d'un canal à QdS : requête[173] du récepteur (REQ_1) et réponse de l'émetteur (ANS_1) ; si elle est positive, la réponse indique le protocole de Transport à mettre en œuvre ;
- le type 2 concerne les échanges entre l'hôte émetteur et son BB local (H_E et BB_A Figure 32) :
  - 2 PDU sont échangés pour la récupération des services et de leurs performances : requête de l'hôte (REQ_2) et réponse du BB local (ANS_2) ; la pré-réservation des ressources est effectuée par le BB local suite à la réception d'un REQ_2 ;

[172] La fonction de répartition du délai de transit.

[173] Indiquant la QdS requise et l'identifiant du flux.

- 2 PDU sont échangés pour la confirmation de réservation et son maintien : requête de réservation (RESV_2) et rafraîchissement (REFR_2) de l'hôte émetteur à son BB local ;
- Note : en cas d'absence de rafraîchissement, la réservation est relâchée ;

– le type 3 concerne les échanges entre les BB (incluant le BB local) :
- 2 PDU sont échangés pour la récupération des services et de leurs performances : requête d'un BB (REQ_3) au BB suivant et réponse (ANS_3) d'un BB au BB local ; la pré-réservation des ressources est effectuée par les BB suite à la réception d'un REQ_3 ;
- 2 PDU sont échangés pour la confirmation de réservation et son maintien : requête de réservation (RESV_3) et rafraîchissement (REFR_3) du BB local à chaque BB distant ;

– le type 4 concerne les échanges entre le BB local et le routeur de bordure par lequel l'hôte émetteur accède à son domaine de souscription (BB_A et R_{A1} Figure 32) :
- 2 PDU sont échangés pour la configuration[174] du routeur : requête de configuration (RESV_4) et rafraîchissement (REFR_4) du BB local au routeur ;

La Figure 33 illustre le déroulement d'une requête de QdS qui aboutit positivement :

– la récupération des classes de services et de leurs performances est effectuée durant la phase 1, en parallèle avec les pré-réservations des ressources pour les classes disponibles ;

– suite au choix d'une classe répondant à la requête de QdS, la confirmation de réservation est réalisée en phase 2 ; le routeur de bordure de l'hôte émetteur est également configuré ;

– la phase 3 illustre le maintien d'une réservation auprès des BB et du routeur de bordure.

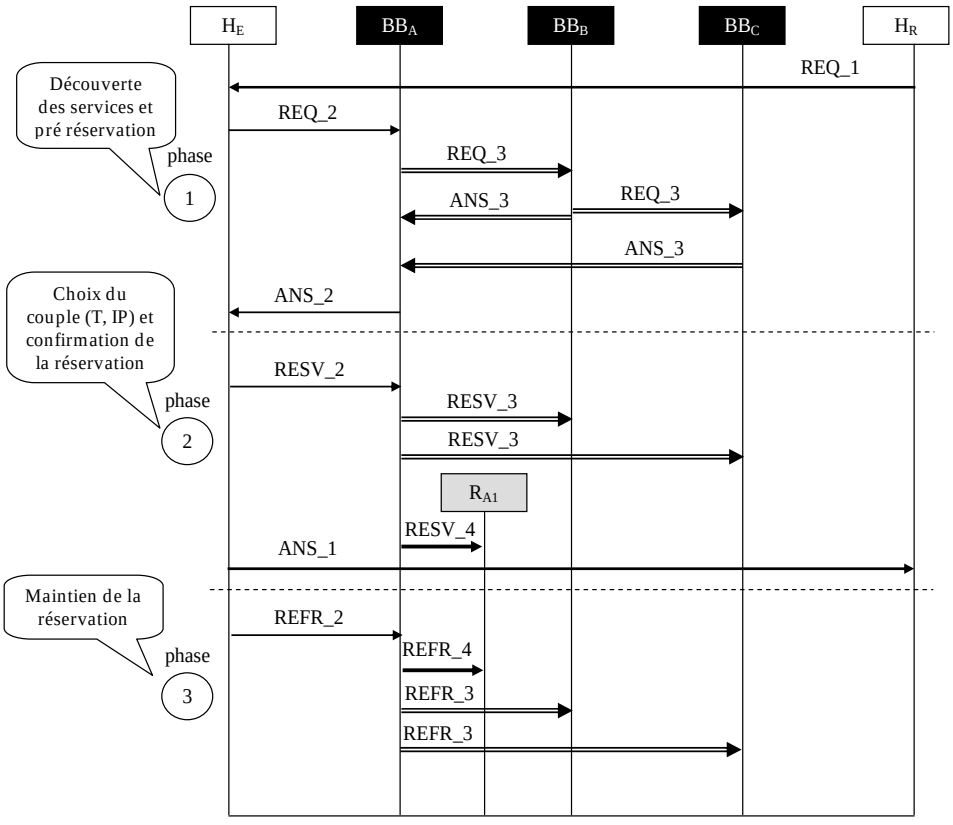


Figure 33 : Echanges de PDU

[174] Classification, *policing* par flux, marquage

2.2.2.3. Identification du prochain BB et des routeurs d'entrée et de sortie

Une signalisation découplée du chemin de donnée pose deux problèmes :

- pour la propagation de la requête au BB suivant, il est nécessaire de l'identifier ;
- un BB doit pouvoir déterminer les deux routeurs de bordure de son domaine par lesquels passe le chemin de donnée.

Dans notre proposition, les deux problèmes se posent lors de la phase 1 de la Figure 33 :

- d'une part, pour les PDU REQ_3, qui sont échangés de BB en BB ;
- d'autre part, pour les PDU ANS_3, qui contiennent les performances de chaque classe de service, entre les routeurs de bordure du tronçon intra domaine.

Notre solution s'appuie sur l'hypothèse que la table BGP est récupérée par les BB dont nous rappelons ci-après la constitution.

Table BGP

La table[175] d'un routeur BGP contient principalement, après filtrage des routes indésirables :

- la liste des réseaux de destination (préfixes IP) accessibles ;
- pour chaque préfixe :
  - l'adresse du prochain routeur (*next-hop*) dans le domaine adjacent ;
  - la liste des AS successivement traversés (*path*) depuis l'AS adjacent jusqu'à l'AS correspondant au réseau visé ;
- pour chaque AS adjacent : l'adresse du prochain routeur voisin (*neighbor*).

La Figure 34 et les Tableau 5 et Tableau 6 permettent d'illustrer ces définitions.

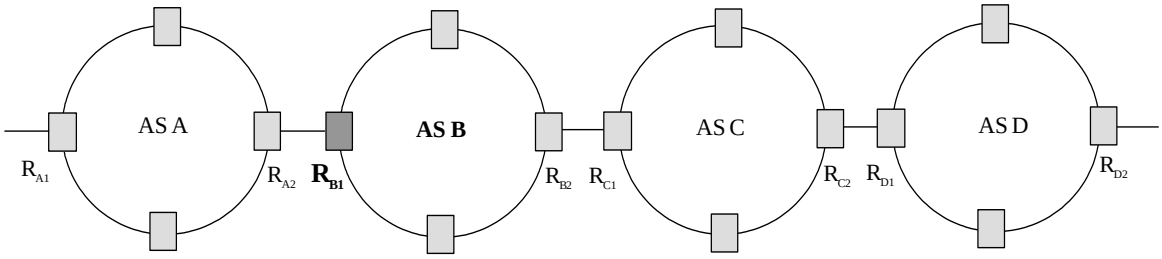


Figure 34 : Exemple d'Internet multi domaines

Le Tableau 5 fournit la vue des *next hop* de la table du routeur R_{B1}.

IP Prefix	Next hop	Path
{préfixes IP} de AS_A	R_{A2}	AS_A
{préfixes IP} de AS_C	R_{C1}	AS_C
{préfixes IP} de AS_D	R_{C1}	AS_C, AS_D

Tableau 5 : Table BGP du routeur R_{B1} de la Figure 34 (vue des *next hop*)

[175] Distribuée sur chaque équipement BGP du domaine.

Le Tableau 6 fournit la vue des *neighbors* du routeur R_{B1} : pour assurer le routage depuis un routeur de bordure d'entrée (par ex. R_{B1}) d'un AS (par ex. AS_B) vers un AS distant (par ex. AS_D), il est nécessaire que R_{B1} connaisse le routeur de bordure voisin (*neighbor*) (ici R_{B2}) qui assure la connectivité avec l'AS adjacent suivant (ici AS_C).

Celui ci peut appartenir au même AS ou être le *next hop*.

AS	Neighbor
AS A	R_{A2}
AS C	R_{B2}

Tableau 6 : Table BGP du routeur R_{B1} de la Figure 34 (vue des *neighbors*).

Solution au problème d'identification du prochain BB

Pour répondre à ce problème, notre proposition est que chaque BB dispose de la liste des couples (AS adjacent, @ du BB de l'AS adjacent).

Recevant un REQ_2 ou REQ_3 (qui identifie en son sein l'@ IP de l'hôte récepteur), un BB consulte la table BGP, extrait le premier AS indiqué dans le *path*, et par suite, identifie le BB (et le routeur d'entrée - *next hop*) de cet AS.

Il est alors en mesure, si nécessaire, de propager la requête au BB suivant.

Solution au problème d'identification des routeurs d'entrée et de sortie d'un domaine

Nous faisons l'hypothèse que les BB accèdent aux tables BGP de leur domaine.

Pour identifier le routeur de sortie, le BB détermine dans la table des *neighbors* du routeur d'entrée, celui qui correspond au premier AS du *path* indiqué dans la table des *next hop*.

Le *next hop* constituant le routeur d'entrée du prochain AS, il peut être transmis au BB suivant dans un REQ_3.

2.2.3. Implémentation et tests expérimentaux

Le protocole de signalisation a été spécifié en UML 2, puis implémenté en JAVA et intégré à l'architecture @IRS++ initiale.

Afin de vérifier l'efficience du système proposé, l'architecture @IRS++/MD a été déployée sur une plate forme d'émulation (*dummynet*) de quatre domaines DiffServ, pour chacun desquels il était possible de modifier le modèle de performance entre l'entrée et la sortie.

Deux types de scénarios de test ont été établis :

- le premier regroupe des tests fonctionnels concernant la découverte d'un service de bout de bout, sa réservation et son maintien, selon huit scénarios conduisant à des situations d'échec et un scénario de succès ;
- le deuxième regroupe des tests sur la qualité de présentation perçue au niveau utilisateur pour des applications de *streaming* vidéo et de visioconférence s'appuyant sur le système de communication proposé.

Ces travaux sont décrits de façon détaillée dans [Auri04a].

V. Conclusion

Les travaux présentés dans cette étape visaient à proposer une architecture de bout en bout offrant des garanties de QdS pour des applications supposées à larges spectres de besoins, distribuée dans un Internet de type DiffServ, d'abord mono domaine, puis multi domaines.

Comparativement aux autres travaux, notre approche était d'une part d'intégrer les niveaux Transport et IP (qui ont chacun un impact sur la QdS), et d'autre part, d'abstraire à l'application la complexité du choix et du paramétrage des services Transport et IP permettant de satisfaire la QdS requise. En cela, elle abordait le problème de la QdS d'un point de vue utilisateur, ignorant des caractéristiques du réseau mais souhaitant minimiser l'utilisation des ressources les plus coûteuses.

Ces travaux nous ont conduit :

- d'abord, dans le cadre du projet @IRS, à proposer, à implémenter et à évaluer les performances d'une architecture offrant une QdS IP par flux dans un environnement DiffServ mono domaine offrant en cœur une QdS par agrégat ;
- ensuite, dans le projet @IRS++, à intégrer un Transport à fiabilité partielle à cette architecture, et à proposer un algorithme de sélection du meilleur couple (Transport/IP) permettant de satisfaire une requête de QdS exprimée en des termes génériques (API), sans que l'application ait à expliciter les services à activer. Cet algorithme s'appuie sur un modèle des performances[176] du service de bout en bout, dont l'origine provient des mesures réalisées dans le projet @IRS, confortées ensuite en simulation (ns-2) ;
- enfin, en préambule au projet EuQoS, à proposer une signalisation pour l'architecture @IRS++, afin que l'approche soit envisageable dans un environnement DiffServ multi domaines. Ce travail nous a en particulier conduit à étendre le modèle de performance pour le contexte du multi domaines.

Les limites de ces travaux se situent à deux niveaux :

- d'une part, au niveau du modèle de performance, qui, au niveau IP, repose sur l'hypothèse d'indépendance en probabilité des paquets d'un même flux, et au niveau Transport, méconnait certains des mécanismes ayant un impact sur le QdS, en particulier le contrôle de congestion ; repousser cette dernière limite fait partie de nos perspectives ;
- d'autre part, au niveau de nos hypothèses de modèle d'Internet multi domaines (supposés totalement homogènes), sur lesquelles repose le protocole de signalisation ; la levée de ces hypothèses et la résolution des problèmes posés en conséquence font partie de notre prospective de recherche, discutée dans le chapitre suivant.

* * *

[176] La fonction de répartition du délai de transit.

CONCLUSION ET PROSPECTIVE

I. Conclusion générale

Ce mémoire a résumé nos contributions aux protocoles et aux architectures de bout en bout visant à offrir, dans l'Internet multi domaines, un service de communication à QdS garantie, pour des applications multi flux, présentant des besoins nouveaux (synchronisation, fiabilité partielle, délai, …) vis à vis des applications initiales de l'Internet.

L'approche adoptée part de l'étude des besoins des applications au niveau de la présentation des données, sachant que le système de communication est imparfait, c'est à dire susceptible de générer des pertes, des délais et des déséquencements. Elle nous conduit ensuite à :

- élaborer un modèle qui exprime les propriétés minimales devant être vérifiées pour que la présentation distante conserve son intelligibilité ;
- étudier les mécanismes à intégrer au niveau du système de communication (niveaux Transport et IP) pour satisfaire ces propriétés ;
- mettre en œuvre et évaluer les solutions proposées.

Les travaux réalisés ont été présentés en trois temps.

▪ Dans l'étape 1, qui considérait un contexte Internet de type *best effort*, nous avons plus particulièrement traité les applications multimédia (audio/vidéo) et les protocoles de Transport ; nous avons successivement :

- étudié les applications multimédia et proposé un modèle d'expression de leur besoin en synchronisation logique (ordre partiel série parallèle) et de leur tolérance aux pertes (taux de perte, nombre maximal de pertes consécutives, …) ;
- proposé un nouveau concept de service et protocole de Transport, à ordre et fiabilité partiels, qui répond aux contraintes exprimées par le modèle ;
- proposé une nouvelle architecture pour les protocoles de Transport multimédia, à multi connexions, permettant de contrôler l'ordre et la fiabilité tant au niveau flux (connexion) qu'au niveau inter flux ;
- contribué à l'évaluation de cette architecture via des modèles OPNET.

▪ Dans l'étape 2, nous avons considéré les applications de simulation interactive distribuées (DIS) et un nouveau contexte Internet, à QdS de type *IntServ* ; ceci nous a conduit à :

- étudier les applications DIS et proposer un modèle d'expression de la cohérence spatiale et temporelle, inhérente à cette famille d'application (classification des PDU DIS avec fiabilité et délai associés à chaque classe) ;
- proposer une architecture de bout en bout (DIS/ATM), qui, à partir d'une caractérisation des entités simulées localement et d'une méthodologie de caractérisation du trafic DIS en termes du TSPEC de RSVP, contrôle l'établissement des réservations RSVP ;
- implémenter l'architecture et mesurer ses performances sur une plateforme nationale.

▪ Dans l'étape 3, nous avons considéré un nouveau contexte Internet, à QdS de type *DiffServ*, en abordant l'intégration des niveaux Transport et IP, puis la signalisation pour la QdS dans le contexte du multi domaines. Les applications ciblées étaient supposées à large spectre de besoins en QdS.

Ceci nous a conduit à proposer :

- un modèle générique d'expression de la QdS permettant d'exprimer une sémantique de la garantie attendue par l'application pour les paramètres requis ;
- un modèle de performance des services DiffServ mono domaine (fonction de répartition du délai de transit), puis sa composition pour l'étendre au multi domaines ;
- un algorithme de sélection automatique du meilleur couple (Transport, IP) satisfaisant les paramètres de QdS et leur sémantique de garantie requis par l'application ;
- un protocole de signalisation (spécifié en UML 2) permettant d'assurer la découverte des services et de leurs performances, puis la réservation des ressources, dans un contexte multi domaines.

En termes d'implémentation et de validation, nous avons :

- implémenté et mesuré les performances d'une première architecture (@IRS) pour un contexte DiffServ mono domaine, assurant un conditionnement du trafic par flux en bordure du domaine. Les mesures, complétées par des simulations ns-2, ont servi à élaborer le modèle de performance des services DiffServ ;
- reconsidéré l'API @IRS, afin qu'elle intègre le modèle générique d'expression de la QdS et l'algorithme de sélection automatique des services (architecture @IRS++) ;
- validé en émulation et implémenté le modèle de performance pour le multi domaines ;
- implémenté le protocole de signalisation pour le multi domaines, intégrer ce protocole à l'API et testé l'architecture complète (@IRS++/MD) pour plusieurs applications multimédia sur une plateforme d'émulation de domaines DiffServ.

Nos présentons maintenant notre prospective pour les années à venir.

II. Prospective

Notre projet de recherche comporte deux parties complémentaires, traitées de façon indépendante, mais ayant vocation à s'intégrer ultérieurement.

Elles concernent :

- à court terme, la signalisation pour la QdS dans l'Internet multi domaines hétérogènes ;
- à plus long terme, les protocoles de Transport configurables dynamiquement et les architectures auto adaptatives pour répondre aux besoins en communication dans le futur contexte de l'Internet, défini comme l'*Internet ambiant*.

1. Signalisation pour l'Internet multi domaines hétérogènes

A l'heure actuelle, les propositions de gestion de la QdS dans l'Internet multi domaines s'appuient sur plusieurs modèles ; le plus générique repose sur le concept de BB, non obligatoirement localisés sur le chemin de donnée. La coopération nécessaire entre les BB impose une signalisation découplée du chemin de donnée au sens défini par NSIS.

Nos travaux d'architecture, incluant la signalisation, pour la QdS dans l'Internet multi domaines, s'inscrivent dans ce modèle et cherchent à satisfaire le besoin en signalisation.

Cependant, ils s'appuient sur l'hypothèse de domaines supposés tous aptes à offrir des services différenciés et à participer au contrôle d'admission à l'échelle du multi domaines, par le biais des BB. En outre, les BB sont supposés uniques au sein de chaque domaine.

Notons que cette hypothèse est adoptée par la quasi totalité des travaux actuels, quel que soit le modèle de gestion de la QdS.

Notre objectif est à présent d'aborder le contexte d'un Internet ***hétérogène*** *suivant un modèle plus réaliste, incluant, par exemple, des domaines soit n'offrant qu'une seule classe de service IP mais sur provisionnés en bande passante, soit aptes à différencier quelques classes, mais sans respecter les règles d'une gestion à l'échelle du multi domaines.*

Dans la suite, nous distinguerons les domaines que nous qualifierons de *non conformes* au modèle de gestion de la QdS inter domaines (DNC), des *domaines conformes* (DC).

Dans un premier temps, nous considérerons le cas d'îlots de DNC, tel qu'illustré Figure 35, c'est à dire tels que les domaines d'extrémité sont des DC.

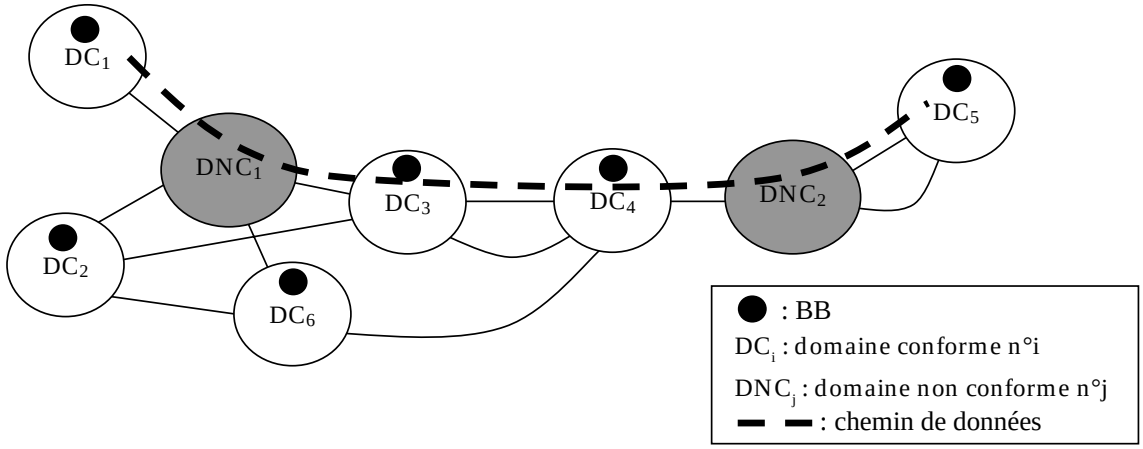


Figure 35 : Chemin hétérogène

La problématique nouvelle est d'assurer la continuité du CAC inter domaines (incluant la découverte des services et de leurs performances) en assurant la propagation de la signalisation entre les BB situés sur les DC amont et aval de chaque DNC, par exemple DC_1 et DC_3 vis à vis de DNC_1 sur la Figure 35.

La présence des DNC repose, avec de nouvelles contraintes, les précédents problèmes :

- d'évaluation des performances intra domaine des DNC : celle-ci doit être effectuée et mise à disposition, depuis des DC ; par exemple Figure 35 : celle de DNC_2 entre DC_4 et DC_5 ;
- de détermination du prochain BB : lorsque le prochain domaine est un DNC, le BB amont doit découvrir le BB d'un DC non adjacent ; par exemple Figure 35 : détermination par le BB du DC_1 du BB de DC_3. Notons que si l'unicité des BB n'est pas vérifié, le problème se pose aussi entre DC adjacents ;
- de contrôle d'admission des liens inter domaines : la difficulté est de gérer le lien d'entrée d'un DC situé en aval immédiat d'un DNC ; par exemple Figure 35 : le lien DNC_1-DC_3.

Notre propos est de traiter ces différents problèmes :

- d'une part, dans un contexte indépendant du type de version de BGP ;
- d'autre part, en proposant nos travaux au groupe NSIS, qui dans sa deuxième phase, a pour objectif de dépasser l'alternative entre signalisations couplée et découplée au/du chemin de donnée.

Dans sa première phase (actuelle), NSIS poursuit une approche de signalisation couplée au chemin de donnée, ce qui est a priori inadéquat pour une signalisation inter BB, mais on doit remarquer qu'en présence d'îlots DNC, cette approche assure naturellement la continuité de la propagation de la signalisation sur le chemin.

De plus, les travaux récents de NSIS [Hanc05b] abordent le besoin en signalisation inter BB via une approche de type COPS (en mode *outsourcing*), autorisant certains équipements NSIS du chemin à détourner les paquets de signalisation vers leur BB, qui, après traitement, les leur renvoie pour qu'ils poursuivent leur route.

Malgré les promesses de ces extensions, certains problèmes demeurent liés à la multiplicité des chemins entre deux extrémités pour des classes de services différentes. Par exemple sur la Figure 35, pour aller de DC_2 à DC_4, une classe de service donnée transitera par DC_3, une autre par DC_6. Dans ces conditions, la notion de signalisation couplée au chemin doit être réexaminée pour assurer que la signalisation suit effectivement le chemin de donnée auxquelles elle est associée.

Nous proposerons une signalisation inter BB qui traite le contexte du multi domaines incluant des DNC, en visant à la compatibilité avec NSIS et ses extensions, et en s'efforçant de répondre aux problèmes posés par la multiplicité possible des chemins inter domaines et celle des BB au sein de chaque domaine.

Nous présentons maintenant la deuxième partie de notre projet de recherche, qui oriente nos travaux à plus long terme.

2. Protocoles de Transport auto configurables et architectures auto adaptatives

Ces toutes dernières années ont vu la multiplication de nouvelles technologies de communication sans fil (WiFi, *Bluetooth*, …) constituant la préfiguration d'une mobilité généralisée des utilisateurs dans l'Internet. Ainsi, il est déjà possible de se connecter à l'Internet depuis plusieurs types de points d'accès (filaire ou sans fil) et via plusieurs sortes de terminaux (station de travail, portable, PDA, …).

Dans les années à venir, la communication devrait être diffusable beaucoup plus largement par une utilisation massive de processeurs embarqués de type capteurs ou actionneurs.

L'Internet sera alors extrêmement diffus, mobile et hétérogène, au moins à sa périphérie, c'est à dire là où se situeront les utilisateurs.

Dans ce contexte, un enjeu majeur des recherches à venir est de permettre aux utilisateurs d'accéder au réseau et à ses services, multiples et différents selon les endroits, en ayant une conscience minimale de leur environnement [Clar05].

Plus précisément, il s'agit que l'utilisateur puisse, quelle que soit sa localisation, accéder à un environnement de communication dont il ignore potentiellement les caractéristiques, mais

qui lui fasse bénéficier des services maximums compatibles, d'une part, avec les ressources disponibles (bande passante, serveurs, ...), et d'autre part, avec ses préoccupations principales, que celles-ci soient explicitées ou non.

L'horizon qui se dessine ainsi est celui d'un *Internet ambiant*, à la fois *ubiquitaire*, *pervasif*[177], et *intelligent* au travers duquel :

- l'utilisateur sera mobile et donc susceptible d'utiliser des ressources en différents endroits, sans devoir considérer son environnement courant ;
- la communication sera diffusable partout, et avec des caractéristiques différentes (en termes de débits, de taux de pertes, …) en fonction des ressources disponibles ;
- le réseau et ses services seront à même de satisfaire, voire d'anticiper, les besoins des utilisateurs, en tirant au maximum partie des capacités de l'environnement (incluant les utilisateurs eux mêmes) dans lequel ceux ci évolueront à chaque instant.

Cette perspective soulève de nombreux défis liés aux problèmes de dynamicité, d'hétérogénéité, de sécurité et de performance ; les solutions imposent de définir de nouveaux services, de nouveaux protocoles et une nouvelle architecture pour l'Internet.

Différentes initiatives, projets et groupes de travail commencent à adresser ces défis, parmi lesquels l'initiative *Wireless-World initiative*[178], le projet américain OXYGEN[179] du MIT, ou encore les projets européens CONTEXT[180], *End to End Reconfigurability* (E²R)[181] *Winner*[182], Ambiants Networks[183]. Au niveau national, le pôle ResCom[184] du GDR ARS[185] a pour objectif général, dans cet horizon, de fédérer des thématiques de recherche en particulier liées aux aspects algorithmiques et structurels, aux outils de modélisation et d'évaluation de performance et aux aspects protocolaires.

Pour aborder la dynamicité, l'*adaptabilité au contexte* apparaît comme le concept majeur qu'auront à définir et à intégrer les solutions proposées [Jean03] [Iked04]. Il s'agit d'une part, de définir de nouveaux modèles de représentation du contexte ciblé[186], et d'autre part, de traiter dynamiquement les évolutions de ce contexte[187].

La *coopération* semble un cadre nécessaire pour traiter l'adaptabilité au contexte. Cette notion existe déjà au travers des solutions dites *pair à pair* développées par exemple au niveau applicatif pour la découverte de fichiers ou celle de route à délai minimal ou permettant de contourner les pares feu ; elle existe aussi au niveau des réseaux, sans fil et *ad hoc*, dont les nœuds sont constitués par les terminaux, susceptibles de servir de routeurs entre les utilisateurs trop éloignés les uns des autres pour communiquer directement.

---

177 Au sens « diffus ».

178 http://www.wireless-world-initiative.org/

179 http://www.oxygen.lcs.mit.edu/

180 http://context.upc.es/index.htm

181 http://www.e2r.motlabs.com/

182 https://www.ist-winner.org/

183 http://www.ambient-networks.org/

184 Réseaux de Communication

185 Architecture Réseau et Système

186 Rôle de l'utilisateur, type d'activité/terminal, caractéristiques du réseau, …

187 Mise à disposition de nouveaux services en cas de changement de rôle, récupération d'un service auprès d'un tiers, …

La *cohérence des coopérations* aux différents niveaux considérés constitue un problème ouvert tant au plan de sa définition que des méthodes permettant de la satisfaire. Par exemple à l'heure actuelle, les solutions pair à pair proposées au niveau applicatif pour concourir à des choix de routes à délai minimal, ignorent les éventuelles solutions (balance de charge par exemple) réalisées aux niveaux sous jacents ; en conséquence, des solutions adoptées à différents niveaux peuvent conduire à des résultats éloignés des espérances du fait de l'ignorance de leurs interactions.

Ce dernier constat souligne les limites du modèle en couches sur lequel repose l'architecture de l'Internet, et que de nombreux travaux tentent d'enrichir par le biais de mécanismes de *cross layering* [Kumw03] [Chen04]. Ce concept de *perméabilité inter couches* vise, par le partage d'informations entre les couches, à améliorer la synergie entre leurs services respectifs.

Notre prospective s'inscrit dans cette vision de l'Internet ambiant.

La problématique que nous souhaitons adresser concerne la qualité des communications, qu'il s'agira de maintenir au plus haut degré en tenant compte à présent d'une dynamicité du contexte à la fois au niveau utilisateur et au niveau du réseau.

Notre approche repose :

- d'une part, sur l'adaptabilité des protocoles de communication aux exigences applicatives et aux contraintes du réseau : dans le prolongement des travaux que nous avons déjà réalisés, notamment au niveau Transport et sur la signalisation, l'adaptabilité ciblée concerne, au delà de l'aspect comportemental (fonctions et mécanismes), l'aspect structurel (architecture interne), les deux étant nécessaires pour appréhender la dynamicité tant comportementale que structurelle du contexte applicatif et réseau ; nous ciblerons de façon prioritaire la couche Transport ;
- d'autre part, sur l'adaptabilité de l'architecture globale, conçue non plus de manière isolée au sein de chaque couche, mais de façon coordonnée afin d'assurer un contrôle et une cohérence des solutions d'adaptabilité sur chaque couche ; nous nous focaliserons de façon prioritaire sur les couches Application et Transport, puis Réseau ; l'approche initiale pour la coordination reposera sur un modèle à base de graphes et de règles de transformation développé au LAAS [Guen04].

En première instanciation du cadre contextuel, nous ciblerons les systèmes de gestion d'opérations d'urgence impliquant la coordination d'intervenants dans des environnements peu ou pas pré-équipés en infrastructure réseau.

Dans ce type de systèmes :

- le réseau est constitué par les participants ;
- les ressources en bande passante sont limitées et variables, en fonction de topologie du réseau, qui évolue avec la progression de la mission ;
- les participants appartiennent à des groupes identifiés et structurés, induisant une hiérarchisation potentielle des collaborations ; cette hiérarchisation est également susceptible d'évoluer ;
- enfin, par la nature des communications (audio, vidéo) ainsi que par le caractère d'urgence de la mission, ces systèmes présentent des besoins en QdS pour les transferts de données ; ces besoins peuvent varier durant l'exercice.

Ce type d'applications fait clairement apparaître les limites des solutions classiques mises en œuvre indépendamment au sein des couches :

- au niveau Réseau, la réservation de ressources à la IntServ/DiffServ n'est pas envisageable, la topologie dynamique du réseau induisant une variabilité des ressources et le risque de pénurie ;
- aux niveaux Application et Transport, les mécanismes d'adaptation ne tiennent pas compte de la structure hiérarchique des groupes coopératifs ; par exemple, le contrôle de congestion tend à assurer une répartition égalitaire de la bande passante incompatibles avec un classement hiérarchique des flux.

Ce cas d'étude permettra de proposer un cadre architectural mettant en jeu les niveaux Application, Transport et Réseau, ainsi qu'une coordination chargée de définir les politiques d'adaptabilité de chaque niveau, à partir d'informations provenant du réseau, des entités de Transport et des utilisateurs impliqués dans l'opération.

A l'intérieur de ce cadre, nous instancierons plus particulièrement une architecture de Transport configurable dynamiquement pour mettre en œuvre la politique destinée à assurer le meilleur service de communication aux applications ; dans la partie coordination, nous traiterons le problème de la définition de la politique de Transport à partir des connaissances issues des trois niveaux.

D'autres instanciations du contexte ciblé qui pourraient tirer partie du cadre architectural proposé sont constituées par les systèmes émergents liés aux nouveaux réseaux de services déployés en *overlay* au dessus de l'Internet, tels que les réseaux d'entreprises avec des flottes nomades, ou encore les réseaux communautaires virtuels.

Ces systèmes sont typiquement constitués pour des groupes contrôlés et éventuellement hiérarchisés d'utilisateurs (le plus souvent) mobiles ; ils reposent sur un réseau de nœuds propriétaires directement administrés et maitrisables, mais s'appuyant sur une infrastructure réseau gérée de manière indépendante, et de ce fait sujette à fluctuation quant à la disponibilité de ses ressources.

Pour ces systèmes, il semble tout à fait approprié de mettre en œuvre, au niveau des nœuds propriétaires, une coordination de l'adaptabilité entre les niveaux Application, Transport et Réseau (d'overlay).

Il nous semble adéquat d'étendre dans un second temps le cadre architectural proposé pour les opérations d'intervention d'urgence à ce type de systèmes.

* * *

REFERENCES BIBLIOGRAPHIQUES

[Alme97] Almesberger W, Salsyan S, Wright S. Quality of service in communication APIs. IFIP 5th International Workshop on Quality of Service (IWQoS'97), New York, 21-23 May 1997.

[Amer94] Amer P, Chassot C, Connolly C, Conrad P, Diaz M. Partial Order Transport Service for Multimedia and other Applications. IEEE ACM Transactions on Networking, October 1994, vol. 2, n° 5.

[Ash05] Ash J, Bader A, Kappler C. QoS-NSLP QSPEC Template. Internet draft, work in progress, October 2005.

[Auri03] Auriol G, Chassot C, Diaz M. Evaluation des performances d'une architecture de communication à gestion automatique de la QdS. 10ème Colloque Francophone sur l'Ingénierie des Protocoles (CFIP'03), Paris, France, Septembre 2003.

[Auri04a] Auriol G. Spécification et implémentation d'une architecture de signalisation à gestion automatique de la QdS dans un environnement IP multi domaines. Thèse de doctorat en Réseaux de l'EDIT, Toulouse, Novembre 2004.

[Auri04b] Auriol G, Chassot C, Diaz M. Architecture de communication à gestion automatique de la QdS en environnement IP à services différenciés. Technique et Science Informatique (TSI), vol. 23, n°9, décembre 2004.

[Auri04c] Auriol G, Chassot C, Diaz M. Toward a signaling architecture in a DiffServ multi-domain environment for per flow end to end QoS. Australian Telecommunications Networks and Applications Conference (ATNAC'04), december 8-10, 2004.

[Bade05] Bader A, Westberg L, Karagiannis G, Kappler C, Phelan T. RMD-QoSM - The Resource Management in DiffServ QoS Model. Internet draft, work in progress, October 2005

[Bert02] Berthou P, Gayraud T, Owezarski P, Diaz M. Multimedia Multi networking: a new concept. Annales des telecommunications, Tome 57, n°7-8, pp.721-750, Juillet-Août 2002.

[Bert91] Berthomieu B, Diaz M. Modeling and Verification of Time Dependent Systems using Time Petri Nets. IEEE Transactions on Software Engineering, 1991.

[Blak98] Blake S, Black D, Carlson M, Davies E, Wang., Weiss W. An Architecture for Differentiated Services. RFC 2475, December 1998.

[Bona00] Bonald T, May M, Bolot J. Analytic Evaluation of RED Performance. IEEE INFOCOM'2000, Tel Aviv, Israel, 2000.

[Brad97a] Braden R, Zhang L, Berson S, Herzog S, Jamin S. Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification. RFC 2205, September 1997.

[Brad97b] Braden R, Zhang L. Resource ReSerVation Protocol (RSVP) -- Version 1 Message Processing Rules. RFC 2207, September 1997.

[Brad98] Braden R, Hoffman D. RAPI: An RSVP Application Programming Interface. Internet Draft, August 1998.

[Bres00] Breslau L, Knightly E, Shenker S, Stoica I, Zhang H. Endpoint Admission Control: Architectural Issues and Performance. ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOM'00), August 28 - September 1, 2000, Stockholm, Sweden.

[Camp94a] Campbell A, Coulson G, Hutchison D. A quality of service architecture, ACM Computer Communication Review, April 1994.

[Camp94b] A. Campbell, G.Coulson, D.Hutchison. Flow Management in a Quality of Service Architecture. 5th IFIP Conference on High Performance Networking, June 27 - July 1, 1994.

[Chan01] Chan K, Seligson J, Durham D, Gai S, McCloghrie K, Herzog S, Reichmeyer F, Yavatkar R, Smith A. COPS Usage for Policy Provisioning (COPS-PR). RFC 3034, March 01.

[Chas95] Chassot C. Architecture de Transport Multimédia à Connexions d'Ordre Partiel. Thèse en Informatique de l'Institut National Polytechnique de Toulouse, Décembre 1995.

[Chas96] Chassot C, Diaz M, Lozes A. From the Partial Order Concept to Partial Order Multimedia Connections. Journal for High Speed Networks, vol.5, n°2, 1996.

[Chas99] Chassot C, Lozes A, Garcia F, Diaz M, Dairaine L, Rojas L. QoS required by a DIS application in a new generation Internet. IEEE Spring Simulation International Workshop, Orlando (USA), March 14-19, 1999.

[Cha00] Chassot C, Lozes A, Garcia F, Diaz M, Dairaine L, Rojas L. Caractérisation et mise en œuvre de la QdS requise par une application DIS distribuée dans un environnement Internet de nouvelle génération. Revue Electronique sur les Réseaux et l'Informatique Répartie (RERIR), Avril 2000.

[Chas02] Chassot C, Garcia F, Auriol G, Lozes A, Lochin E, Anelli P. Performance analysis for an IP differentiated services network. International Communication Conference (ICC'02), New York, USA, May 2002.

[Chas03] Chassot C, Auriol G, Diaz M. Automatic management of the QoS within an architecture integrating new Transport and IP services in a DiffServ Internet. 6th IFIP/IEEE International Conference on Management of Multimedia Networks and Services (MMNS'03), Belfast, Ireland, September 7-10, 2003.

[Chen04] Chen J, Tiejun L, Zheng H. Cross-layer design for QoS wireless communications. International Symposium on Circuits and Systems (ISCAS'04). Vancouver, Canada, May, 23-26, 2004.

[Cher89] Charinton DR, Williamson CL. VMTP as the Transport layer for high-performance distributed systems. IEEE Communications Magazine, pages 37-44, June 1989.

[Ches87] Chesson G. The Protocol Engines Project. Unix Rev, September 1987.

[Ches89] Chesson G. XTP/PE Design Considerations. IFIP International Workshop on Protocols for High-Speed Networks, Rüschlikon, May 9-11, 1989.

[Clar87] Clark D.D, Lambert M.L, Zhang L. NETBLT: a bulk data transfer protocol. RFC 998, March 1987.

[Clar05] Clark D, Partridge C, Braden R, Davie B, Flyod S, Van Jacobson, Katabi D, Minshall G, Ramakrishnan K, Roscoe T, Stoica I, Wroclawski, Zhan L. Making the world (of communications) a different place. ACM SIGCOMM Computer Communication review, vol. 35, n°2, July 2005.

[Cono94] Connolly T, Amer P, Conrad P. An Extension to TCP: Partial Order Service. IETF RFC 1693, November 94.

[Corte03] Cortes G, Fiutem R, Cremonese P, D'Antonio S, Esposito M, Romano S.P, Diaconescu A. CADENUS: Creation and Deployment of End User Services in Premium IP Networks. IEEE Communications Magazine, vol. 41, n°1, January 2003.

[Cris02] Cristallo G, Jacquenet C. An Approach to Inter-domain Traffic Engineering. Proceedings of the 18th World Telecommunications Congress (WTC'02), Paris, France, September 2002.

[Cris03] Cristallo G, Jacquenet C Providing Quality of Service Indication by the BGP-4 Protocol: the QOS_NLRI attribute. Internet draft, June 2003.

[Dant93] Danthine A, Bonaventure O. From Best Effort to Enhanced QoS, Technical report of the CIO project, July 1993.

[Delg93] Delgrossi L, Sandvoss J. The BERKOM-II Multimedia Transport System. August. 1993.

[Diaz94] Diaz M, Lozes A, Chassot C, Amer P. Partial Order Connections. A new concept for High Speed and Multimedia Services and Protocols. Annals of Telecommunications, May-June 94, vol. 49, n°5-6

[Diaz95] Diaz M, Drira K, Lozes A, Chassot C. On the Definition and Representation of the Quality of Service for Multimedia Systems. 6th IFIP Conf. on High Performance Networking, 1995.

[Durh00] Durham D, Boyle J, Cohen R, Herzog S, Rajan R, Sastry A. The COPS (Common Open Policy Service) Protocol. RFC 2748, Juanuary 2000.

[Eber98] Eberhardt R, Ruess C. Communication Application Programming Interfaces with Quality of Service Support. 1st IEEE Int. Conference on ATM, Colmar, France, 22-24 June 1998.

[Enge03] Engel T, Granzer H, Koch B.F, Winter M, Sampatakos P, Venieris I.S, Husmann H, Ricciato F, Salsano S. AQUILA: Adapative Resource Control for QoS Using an IP-based Layered Architecture. IEEE Communications Magazine, vol. 41, n°1, January 2003.

[Expo03] Exposito E. Specification and implementation of a QoS oriented Transport protocol for multimedia applications. Thèse de Doctorat de l'Institut National Polytechnique de Toulouse, spécialité Réseaux et Télécommunications, Décembre 2003.

[Expo04] Exposito E, Sénac P, Diaz M. UML-SDL modelling of the FPTP QoS oriented transport protocol. International Multi-Media Modelling Conference (MMM'04), Brisbane, Australia, January 5-7, 2004.

[Fall96] Fall K., Floyd S. Simulation-based Comparisons of Tahoe, Reno, and SACK TCP. Computer Communication Review, V. 26 N. 3, July 1996, pp. 5-21

[Floy99] Floyd S, Fall K. Promoting the Use of End-to-End Congestion Control in the Internet. IEEE/ACM Transactions on Networking, Vol.7, n° 4, August 1999.

[Four96] Fournier M, Chassot C, Diaz M, Lozes A. Multimedia Partial Order Transport Architecture: Design and Implementation. Protocol for High Speed Network (PfHSN'96). Sophia Antipolis, November 22-23, 1996.

[Four97a] Fournier M. Réalisation et évaluation de performances d'un protocole de Transport multimédia à ordre partiel. Thèse en informatique de l'Université P. Sabatier. Mars 1997.

[Four97b] Chassot C, Diaz M, Lozes A. Performance evaluation of a partial order connections. High Performance Networking (HPN'97). New-York (USA), 28 Avril - 2 Mai 1997.

[Four97c] Fournier M, Chassot C, Diaz M, Lozes A. Evaluation comparative des performances d'une connexion d'ordre partiel et d'une connexion multimédia d'ordre partiel. 6ème Colloque Francophone sur l'Ingénierie des Protocoles (CFIP'97), Liège, Belgique, 29 septembre - 2 octobre 1997.

[Fred01] Fredj SB, Oueslati-Boulahia S, Roberts JW. Measurement-based Admission Control for Elastic Traffic. 17th International Teletraffic Congress (ITC'01), Salvador, Brazil, December 2001 September 2001.

[Fu05] Fu X, Schulzrinne H, Bader A, Hogrefe D, Kappler C, Karagiannis G, Tschofenig H, Van den Bosch S. NSIS: A New Extensible IP Signaling Protocol Suite. IEEE Communications Magazine, Internet Technology Series, October 2005.

[Fuze03] Füzesi P, Németh K, Borg N, Holmberg R, Cselényi I. Provisioning of QoS enabled inter-domain services. Computer Communications, vol. 26 n°10, June 2003.

[Gane05] Ganesh AJ, Key P, Polis P, Srikant R. Congestion Notification and Probing Mechanisms for Endpoint Admission Control. To appear in the IEEE/ACM Transactions on Networking, 2005.

[Garc01] Garcia F, Chassot C, Lozes A, Diaz M, Anelli P, Lochin E. Conception, implementation and evaluation of a QoS based architecture for an IP environment supporting differentiated services. 8th International Workshop on Interactive Distributed Multimedia Systems (IDMS'01), Lancaster, United Kingdom, September 2001.

[Garc02b] Garcia F, Auriol G, Chassot C, Lozes A, Lochin E, Anelli P. Conception, implémentation et mesures de performance d'une architecture de communication à QdS garantie dans un domaine IPv6 à services différenciés. 9ème Colloque Francophone sur l'Ingénierie des Protocoles (CFIP'02), Montréal, Canada, Mai 2002.

[Garc02a] Garcia F. Conception, implémentation et mesures des performances d'une architecture de communication de bout en bout à QdS en environnement Internet nouvelle génération. Thèse de Doctorat en Informatique de l'école doctorale Système (EDSYS). Décembre 2002.

[Gior03] Giordano S, Salsano S, Van den Berghe S, Ventre G, Giannakopoulos D. Advanced QoS provisioning in IP Networks: the European Premium IP projects. vol. 41, n°1, Jan. 2003.

[Gode01] Goderis D, T'joens Y, Jacquenet C, Memenios G, Pavlou G, Egan R, Griffin D, Georgatsos P, Georgiadis L, Van Heuven P. Service Level Specification Semantics, Parameters and negotiation requirements. Internet draft, June 2001.

[Gopa95] Gopalakrishna G, Parulkar G. A framework for QoS guarantees for multimedia applications within end system. Jahrestagung der deutschen Gesellschaft fur Informatik und der Schweizer Informatiker Gesellschaft, Zurich, Switzerland, September 18-20, 1995.

[Guen04] Guennoun K, Drira K, Ghostine R. An efficient algorithm for extended graphs transformation: description, analysis and evaluation. Rapport LAAS-CNRS. Nov. 2004.

[Hanc05a] Hancock R, Karagiannis G, Loughney J, Van den Bosch S. Next Steps in Signaling (NSIS): Framework. IETF RFC 4080, June 2005.

[Hanc05b] Hancock R, Kappler C, Quittek J, Stiemerling M. A Problem Statement for Path-Decoupled Signalling in NSIS. Internet Draft, work in progress, May 2005.

[Hand03] Handley M, Floyd S, Padhye J, Widmer J. TCP Friendly Rate Control (TFRC): Protocol Specification RFC 3448, January 2003.

[Hass02] Hassanein, H.S.; Haiqing Chen; Mouftah, H.T. Providing packet-loss guarantees in DiffServ architectures. IEEE International Performance, Computing, and Communications Conference (IPCCC'02), Phoenix, Arizona 3-5 April 2002

[Hein99] Heinanen J, Baker F, Weiss W, Wroclawski J. An Assured forwarding PHB. RFC 2597, June 1999.

[Howa05] Howarth MP, Flegkas P, Pavlou G, Wang N, Trimintzios P, Griffin D, Griem J, Boucadair M, Morand P, Asgari A, Georgatsos P. Provisioning for interdomain quality of service: the MESCAL approach. IEEE Communications Magazine, Vol. 43, n°6, June 2005

[Sdt 1278.1] IEEE Standard for Distribued Interactive Simulation – Application Protocol, IEEE standart 1278.1,1995

[Sdt 1278.2] IEEE Standard for Distribued Interactive Simulation – Communication Services and Profiles, IEEE standart 1278.2,1995

[Ihid02] Ihidoussene D, Lecuire V, Lepage F. Un transport à fiabilité partielle par codes de Reed Solomon pour les flux vidéo MPEG temps réels. 9ème Colloque Francophone sur l'Ingénierie des Protocoles CFIP'02, Montréal, 27-30 mai 2002.

[Iked04] Ikeda H, Deng H, Niu Z, Tanabe S. Context-aware quality of service control in session based IP networks. 5th International Symposium on Multi-Dimensional Mobile Communications Proceedings. Joint Conference of the 10th Asia-Pacific Conference, August- September 2004.

[Jaco99] V. Jacobson, K. Nichols, K. Poduri. An Expedited Forwarding PHB. RFC 2598, June 1999.

[Jean03] Jean K., Yang K., Galis A. A Policy Based Context-aware Service for Next Generation Networks. 8th London Communication Symposium, September 8-10, 2003, London.

[Jesp97] Jespen T, Wright SA. A java syntax mapping to the native ATM services. ATM Forum SAA Working group, September 1997.

[Kara04] Karagiannis G, Bader A, Pongracz G, Csaszar A, Takacs A, Szabo R, Westberg L. RMD - a lightweight application of NSIS, Networks 2004, pp. 211 - 216, ISBN 3-8007-2840-0, 2004.

[Kohl02] Kohler E, Handley M, Floyd S. Datagram Congestion Control Protocol (DCCP). IETF Internet-draft, draft-kohler-dcp-03.ps.

[Kumw03] Kumwilaisak W, Hou YT, Zhang Q, Zhu W, Jay Kuo CC, Zhang YQ. A cross-Layer quality-of-service mapping architecture for video delivery in wireless networks. IEEE Journal on Selected Areas in Communications, Vol. 21, n° 10, Dec. 2003.

[Lecu01] Lecuire V, Lepage F, Kammoun K. Enhancing Quality of MPEG Video through Partially Reliable Transport Service in Interactive Application. IFIP/IEEE International Conference on Management of Multimedia Networks and Services (MMNS'01), Chicago IL, USA, October 29 - November 1, 2001.

[Litt90] Little T, Ghafoor A. Synchronization and Storage Models for Multimedia Objects. IEEE J on Selected Areas in Communications, 8(3), 413-427, April 1990.

[Mann05] Manner J, Karagiannis G, McDonald A, Van den Bosch S. NSLP for Quality-of-Service signaling. Internet draft, work in progress, october 2005

[Mant04] Mantar H, Hwang J, Okumus I, Chapin S. A scalable model for inter bandwidth broker resource reservation and provisioning. IEEE Journal on Selected Areas in Communications, Vol. 22, Issue 10, Dec. 2004.

[Mara96] Marasli R, Amer P, Conrad P. Retransmission-based partially reliable transport service: an analytic model. IEEE INFOCOM 96, San Francisco, March 96.

[Mara97] Marasli R, Amer P, Conrad P. An analytic study of partially ordered transport services Computer Networks and ISDN Systems, 29(6), may 97.

[Math96] Mathis M, Mahdavi J, Floyd S, Romanow A. TCP Selective Acknowledgement Options. RFC 2018, Ocotber 1996.

[Myko03] Mykoniati E, Charalampous C, Georgastos P, Damilatis T, Godersi D, Trimintzios G, Pavlou G, Griffin D. Admission Control for Providing QoS in DiffServ IP Networks: the TEQUILA Approach. IEEE Communications Magazine, vol. 41, n°1, January 2003.

[Nahr96] Nahrstedt K, Smith J. Design, Implementation and experiences of the OMEGA end-point architecture. IEEE Journal on Selected Areas in Communications, Vol.14, Sept. 1996.

[Nich99] Nichols K, Jacobson V, Zhang L, A Two-bit Differentiated Services Architecture for the Internet. RFC 2638, July 1999.

[Nord89] Nordmark E, Cheriton D. R. Experiences from VMTP: How to achieve low response time, Protocols for High-Speed Networks, H. Rudin and R. Williamson (Ed), Elsevier Science Publishers B.V. (North-Holland), IFIP 1989.

[Owez98] Owezarski P, Chassot C, Diaz M. A Time Efficient Architecture For Multimedia Applications. IEEE Journal on Selected Areas in Com. (JSAC), April 1998, vol.16, n°3.

[Pan00] Pan P.P, Hahne E, Schulzrinne H. BGRP: Sink-Tree-Based Aggregation for Inter-Domain Reservations. Journal of Communications and Networks, vol. 2, no. 2, June 2000.

[Podu98] Poduri K, Nichols K. Simulation Studies of Increased Initial TCP Window Size. RFC 2415, September 1998.

[Pujo04] Pujolle G (director). Policy based network management. Special issue of the Annals of Telecommunications. Tome 59, n°1-2, january-february 2004.

[QSDT02] QBone Signaling Design Team. Final Report - state of the SIBBS protocol at the time the design team concluded. http://qbone.internet2.edu/bb/index.shtml, June 2002.

[Roja99] Rojas L, Chaput E, Dairaine L, Sénac P, Diaz M. Video Transport Over Partial Order Connections. Computer Networks, vol. 31, issue 7, April 1999.

[Sals03] Salsano S, Winter M, Miettinen N. The BGRP Plus Architecture for Dynamic Inter-Domain IP QoS. First International Workshop on Inter-Domain Performance and Simulation (IPS'03). Salzburg, Austria, 20-21 February, 2003.

[Schu05] Schuzrinne H, Hancock R. GIST: General Internet Signaling Transport. Internet draft, work in progress, September 2005.

[Schw90] Schwaderer W.D. XTP in VLSI: Protocol Decomposition for ASIC Implementation. Proceeding on the 15th Conference on Local Area Networks, IEEE 1990.

[Séna94] Sénac P, Diaz M, de Saqui-Sannes P. Toward a formal specification of multimedia synchronization scenarios. Annales des Télécommunications, vol.49, n°5-6, Mai-Juin 1994.

[Séna96] Sénac P, Diaz M, Léger A, de Saqui-Sannes P. Modeling logical and temporal synchronization in Hypermedia systems. IEEE Journal on Selected Areas in Communications, Volume 14, Number 1, January 1996.

[Shen97] Shenker S, Partridge C. Specification of Guaranteed Quality of Service. RFC 2212, September 1997.

[Shieh01] Shieh CK, Miao YB, Wang CY. A transparent deployment method of RSVP-aware applications on UNIX. 9th IEEE International Conference On Networks (ICON'01), Bangkok, Thailand, October 10-12, 2001

[Shih04] Shih CH, Liao CC, Shieh CK, Huang WS. A transparent QoS mechanism to support IntServ/DiffServ networks. 1st IEEE Consumer Communications and Networking Conference, (CCNC'04).Las Vegas, USA, 5-8 Jan. 2004.

[Stew00] Stewart R, Xie Q, Morneault K, Sharp C, Schwarzbauer H, Taylor T, Rytina I, Kalla M, Zhang L, Paxson V. Stream Control Transmission Protocol. IETF, RFC 2960, October 2000.

[Tene94] TENET group. Recent and Current Research. University of California, Berkeley, USA, April, 1994.

[Terz99a] Terzis A, Ogawa J, Tsui S, Wang L, Zhang L. A prototype Implementation of the Two-Tier Architecture for Differentiated Services, 5th IEEE Real-Time Technology and Applications Symposium (RTAS99) Vancouver, Canada, June 2-4, 1999.

[Terz99b] Terzis A, Ogawa J, Tsui S, Wang L, Zhang L. A two-tier resource management model for the Internet. Proceedings of Global Internet 99, December 1999.

[Tran02] Trang Nguyen TM, Boukhatem N.; Doudane, Y.G.; Pujolle, G. COPS-SLS: a service level negotiation protocol for the Internet. IEEE Com. Magazine, Vol. 40, n°5, May 2002.

[Tran03] Trang Nguyen TM, Boukhatem N.; Doudane, Y.G.; Pujolle, G. COPS-SLS usage for dynamic policy-based QoS management over heterogeneous IP networks. IEEE Communications Magazine, Vol. 17, n°3, May-June 2003.

[Trim01] Trimintzios P, Andrikopoulos I, Pavlou G et al. A management and control architecture for providing IP differentiated services in MPLS-based networks. IEEE Communications Magazine, Vol. 39, n°5, May 2001.

[Trim02] Trimintzios P, Flegkas P, Pavlou G et al. Policy-based network dimensioning for IP differentiated services networks. IEEE Workshop on IP Operations and Management (IPOM'02), Dallas, USA, Nov 2002.

[Tsen05] Tsenov T, Tschofenig H, Fu X, Aoun C. GIST state machine. Internet draft, work in progress, October 2005.

[Walt83] Walter B. Timed petri-nets for modeling and analysing protocols with time. Proceedings of PSTV, III, IFIP, 1983.

[Wang00] Wang P.Y, Yemini Y, Florissi D. QoSME: toward QoS management and guarantees. International Conference on Communication Technology (ICCT'00), Beijing, China, August 21-25, 2000.

[Wats89] Watson R.W. The Delta-t Transport Protocol: Features and Experience. Protocols for High-Speed Networks. H. Rudin and R. Williamson (Ed), Elsevier Science Publishers B.V. (North-Holland), IFIP 1989.

[West02] Westberg, L. Resource Management in Diffserv (RMD): A Functionality and Performance Behavior Overview. IFIP Proceeding Protocols for High Speed Networks (PFHSN'02). Berlin, Germany, April 22 - 24, 2002,

[Wroc97] Wroclawski J. Specification of the Controlled-Load Network Element Service. RFC 2211, September 1997.

[Xiao02] Xiao L, Lui K.-S., Wang J, Nahrstedt K. QoS extension to BGP. 10th IEEE International Conference on Network Protocols (ICNP'02), Paris, France; 12-15 November 2002

[Xiao04] Xiao L, Lui K.-S., Wang J, Nahrstedt K. Advertising interdomain QoS routing information. IEEE Journal on Selected Areas in Communications Vol. 22, n°10, Dec. 2004.

[Yan01] Cheng Yan; Gu Boxuan; Ma Xiaojun; Gu Guanqun. Improving on service-customizing for applications in PGPS-based IntServ/RSVP architecture. 2001. International Conference on Computer Networks and Mobile Computing. Los Alamitos, CA, 16-19 Oct. 2001.

[Yang05] Yang J, Ye J, Papavassiliou S. A flexible and distributed Architecture for adaptive End-to-End QoS Provisioning in Next Generation Networks. IEEE Journal on Selected Areas in Communications, vol. 23, n°2, February 2005.

* * *