République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE KASDI MERBAH, OUARGLA

Faculté des Sciences, Technologie et Sciences de la Matière - STSM

Département de Mathématiques & Informatique

N° d'ordre :……
N° de Série :……

Mémoire

En vue de l'obtention du diplôme de

Magistère en Informatique

(Option: Réseaux et Systèmes d'Information Multimédia)

Présenté par:

Mohammed Salim KORICHI

THEME

QoS par flux dans un environnement multi-domaine et hétérogène

Dirigé par :
Dr. : Driss KORICHI

Soutenu devant le jury:

Dr. Said Moahmed Said (MCA) - Université d'Ouargla	**Président**
Dr. KORICHI Driss (MCA) - Université d'Ouargla	**Rapporteur**
Dr. Belattar Brahim (MCA) – Université de Batna	**Examinateur**
Dr. Ahmed KORICHI (MCA) – Université d'Ouargla	**Examinateur**

Remerciements

Je tiens tout d'abord à remercier mon encadreur, le Dr. Korichi Driss, Maître de Conférences à l'université d'Ouargla, qui, après l'inspiration et la guidance de Dieu, a eu le mérite d'orienter ce travail académique qui traite sujet très moderne dans le domaine de réseaux informatiques. Son soutien et ses orientations m'ont aidé à prendre la bonne route et à achever ce travail.

Je tiens également à exprimer mes remerciements au Dr. Mellouk Abdelhamid, Maître de Conférences à l'Université Paris Val de Marne, pour ses conseils et ses orientations intéressantes.

Je tiens également à remercier les membres du jury qui ont accepté d'examiner et de statuer sur mon travail:

Dr. Said Moahmed Said qui m'a fait l'honneur de présider ce jury de mémoire ;

Dr. Belattar Brahim et Dr. Ahmed Korichi qui ont accepté d'être les examinateurs de mon mémoire et pour l'intérêt qu'ils ont porté à ce travail.

Je suis très reconnaissant à tous mes enseignants de l'école doctorale RSIM à l'université Kasdi Merbah de Ouargla. Je remercie en particulier monsieur Djoudi Mahiéddine, Maître de Conférences à l'université de Poitiers, le Professeur Ben Mohammed Mohamed de l'université de Constantine, le Professeur Bilami Azzeddine, monsieur Zidat Samir Maitre de conférences à l'université de Batna, madame le Docteur Laalam Fatima Zohra Responsable de l'école et monsieur Herrouz Hakim enseignant à l'université d'Ouargla, pour leurs efforts pédagogiques et administratifs qui contribuent chaque jour au succès retentissant de notre école.

J'adresse aussi mes sincères remerciements à tous mes collègues d'étude et en particulier : Hmida, Salah et Hocine pour les discussions scientifiques qui nous ont enrichis et pour leur aide et leur appui durant cette période cruciale de recherche.

Je remercie sincèrement tous mes collègues de travail : Hmida, Abdlatif, Omar, Hocine, Okba, Nassima, Assia et en particulier mon directeur, monsieur Hadji Mohamed Salah pour m'avoir encouragé et aplani tant d'obstacles au cours de la phase de recherche.

Je tiens à remercier tout le staff du département de mathématique et informatique, plus particulièrement le chef de département monsieur Assila Mustapha.

Je remercie chaleureusement mon père et ma mère pour leurs prières qui m'ont accompagné tout le long de mes études, et tous mes frères et mes proches.

Je tiens à remercier vivement mon épouse qui m'a supporté avec notre petite Khadidja durant toute la période d'étude théorique et pratique.

Table de Matières

Liste de figures

Liste de tableaux

Introduction générale

L'internet, dans sa forme native, est vu comme un ensemble de ressources partagées entre les utilisateurs où chacun peut injecter tant de paquets au plus vite que possible sans aucun contrôle de la part du réseau. Au sein du réseau, les paquets de différentes applications sont servis à l'ordre d'arrivée (FIFO : First In, First Out). En cas de congestion, les paquets seraient détruits sans différenciation. C'est ce qu'on appelle le service Best-Effort.

L'utilisation de l'internet se restreignait à la consultation des pages web, le transfert de fichier et la messagerie électronique. Mais, l'évolution des technologies de la communication et de l'informatique a fait apparaitre de nouvelles applications distribuées et aux contraintes de temps comme la voix sur IP, la vidéo à la demande, la visioconférence, les jeux interactifs, la simulation interactive distribuée, etc. Ces applications, et à l'inverse des applications classiques d'internet, imposent de nouvelles contraintes sur la qualité de service exprimées en termes de bande passante, délai de transfert et taux de perte de paquet. Malheureusement, Internet n'a pas été conçu nativement pour répondre à ces contraintes.

Pour faire face à ces nouveaux défis, plusieurs travaux de recherches et d'expérimentation pratiques ont été réalisés. On les classe globalement sur deux axes : *l'allocation de ressources* et *optimisation de performances*. Ces deux notions seront pleinement expliquées dans la partie suivante « contexte général et problématique ».

Ce mémoire rentre dans ce cadre des solutions de QoS (Quality Of Service ou qualité de service QdS en français). Il traite plus particulièrement les problèmes de qualité de services dans les réseaux et les solutions proposées. Il présente une synthèse de l'état de l'art avant de détailler des propositions et des simulations pour répondre à cette problématique.

Dans le premier chapitre, nous avons présenté les technologies et les architectures définies pour assurer la QoS à l'intérieur d'un domaine « Intra-domaines ». Nous avons présenté les architectures IntServ et DiffServ et les mécanismes utilisés pour la gestion de trafic : classification, conditionnement, contrôle d'admission, etc. Aussi nous avons parlé sur le MPLS TE (MultiProtocol Label Switching Traffic Engineering) et le PBNM (Policy Based Network Management) utilisés pour l'optimisation de performance. Une grande partie du chapitre est consacrée au routage à QoS où nous avons cité les défis rencontrés et les solutions proposées. Nous avons donné plus d'importance aux algorithmes proposés pour résoudre le problème MCP (Multi-Constraint Path problem) et au routage inductif basé sur l'intelligence artificielle.

Le deuxième chapitre a été consacré aux solutions se situant entre les domaines « inter-domianes ». Il traite le protocole de routage inter-domaine BGP (Border Gateway Protocol), et le Bandwidth Broker qui est, entre autre, responsable de la communication avec les autres domaines. Ce chapitre traite également les protocoles de signalisation SIP (Session Initiation Protocol), SIBBS (Simple Inter-domain Bandwidth Broker Protocol) et NSIS (Next Steps In Signaling). A la fin du chapitre, nous avons éclairé les notions relatives aux nouvelles extensions de l'MPLS TE qui spécifient la coopération entre les domaines MPLS afin de calculer un chemin inter-domaine.

Le système européen « EuQoS » sera présenté succinctement dans le troisième chapitre. Il décrit brièvement le nouveau modèle de service, l'architecture générale du système, les composants essentiels ainsi que le schéma de contrôle d'admission.

Le quatrième chapitre sera complètement consacré à notre contribution personnelle. Tout d'abord et suite aux solutions inter-domaines proposées au deuxième chapitre, ou nous avons exposé une brève présentation de l'architecture Aquila et les spécifications de ses classes (descripteur de trafic, les objectifs de QoS, contrôle d'admission, etc.). Nous démontreront par simulation le mauvais effet de l'agrégation d'un flux en une classe qui n'est pas conçue pour lui même s'ils ont les mêmes objectifs de QoS. Nous avons utilisé le fameux simulateur NS2. Dans la deuxième partie du chapitre, nous avons exposé notre contribution liée au projet EuQoS. Il s'agit d'une extension au projet pour traiter les flux de nature inconnue.

La conclusion générale du présent mémoire, résume notre travail et rappelle les notions importantes. Nous exposerons également les travaux à faire comme perspectives.

Contexte général et problématique

Contexte général

Comme nous l'avons expliqué dans l'introduction générale, et pour faire face aux nouveaux défis liés aux problèmes de QoS, plusieurs travaux de recherches et d'expérimentation pratiques ont été réalisés. On les classe globalement sur deux axes : l'allocation de ressources et l'optimisation des performances.

Sous l'allocation de ressources, on distingue les problèmes de paquets retardés ou supprimés à cause de l'insuffisance de ressources. A ce propos, deux architectures ont été standardisées : Intégration de services (IntServ) et Différenciation de service (DiffServ). Elles introduisent de nouveaux modèles de services et Frameworks de gestion de ressources : réservation, classification, conditionnement, etc. IntServ traite chaque flux individuellement en y faisant une réservation tout au long du chemin de flux durant la période de la connexion. Alors que DiffServ regroupe les flux en classes et définit un traitement spécifique pour chaque classe. En d'autre terme, IntServ est orienté flux et DiffServ est orienté classe de flux.

L'autre axe de travaux, l'optimisation des performances, vise la question de contrôle des flux dans le réseau de manière très efficace de telle sorte qu'on répond à autant de demandes de QoS mais par le moindre coût. La lacune d'Internet dans ce contexte provient du paradigme « datagramme ». Dans ce paradigme, les protocoles de routage interne IGP (RIP, OSPF, IS-IS, IGRP, etc.) orientent tous les paquets ayant le même préfix d'adresses de destination vers la même route. Le routage qui se base sur l'adresse de destination (destination-based routing) ne permet pas de partager les flux de même destination sur plusieurs routes. Ceci conduit à une mauvaise exploitation de ressources. La technologie MPLS TE, la gestion de réseau par politique (PBNM) et le routage à QoS sont des solutions proposées pour combler cette lacune et permettront de tenir compte des autres facteurs dans le choix de la route.

La technologie MPLS est basée sur la commutation de « labels » qui consiste à insérer aux données « un label » à l'entrée de réseau entre la couche IP et la couche de liaison de données, et à acheminer les données en se basant seulement sur le contenu du label. Dans le cas du MPLS TE, pour sélectionner la bonne route de flux, MPLS TE ajoute la possibilité de prendre en compte d'autres informations sur l'état des liens, la charge actuelle, la charge de futur, les pannes, métrique TE etc. Certaines de ces informations sont échangées par le protocole OSPF TE ou IS-IS TE, et d'autres sont introduites par l'administrateur de réseau.

La gestion de réseau par politique (PBNM) permet, dans notre contexte, une meilleure exploitation des ressources des réseaux IP. Dans cette technique, on centralise la décision d'admission, de réservation et de routage en une entité dédiée connaissant bien l'état du réseau et dotée d'une base de politiques (règles). La base de politiques comprend les règles de gestion du réseau prenant en compte la QoS, la sécurité, etc.

Le routage à QoS est mécanisme de routage par lequel les routes sont déterminées en connaissant, à la fois, les ressources disponibles du réseau et la QoS requise par le flux. Le problème basique de routage à QoS est tout d'abord comment collecter, mesurer, et diffuser les informations pertinentes sur l'état du réseau (bande passante disponible, délai, etc.) ; et puis comment calculer la route «faisable ou optimale» à partir de ces informations en réduisant la complexité des algorithmes en temps et en espace.

D'autre part, l'Internet est un ensemble de domaines interconnectés. La connexion entre deux ou plusieurs utilisateurs Internet traverse éventuellement plusieurs domaines. Rappelons d'abord, qu'un domaine est l'interconnexion d'un ensemble de nœuds formant un système autonome dit AS, qui est interconnecté avec les autres domaines par les nœuds dites de bordure. Chaque domaine est administré par une seule entité (généralement par un opérateur réseau). Pour des raisons de sécurité et de marketing (compétitivité), les opérateurs ne permettent pas la propagation des informations sur la topologie et les ressources disponibles de ses domaines. Ce qui met un obstacle devant l'application des précédentes architectures et mécanismes pour sélectionner un chemin de bout-en-bout satisfaisant la QoS demandée, et rend nécessaire l'extension des solutions existantes pour réaliser la QoS inter-domaine.

Les solutions proposées traitent la sélection de route inter-domaine et la signalisation. Dans ce contexte, la technologie MPLS est l'objet de nouvelles extensions (normalisées ou encore en version Draft) qui spécifient la manière de la coopération entre les domaines afin de sélectionner la route inter-domaine répondant au besoin de QoS.

En ce qui concerne, la signalisation inter-domaine, parmi les solutions proposées on cite : NSIS et SIBBS. Le Framework NSIS se démarque par sa capacité de signaler plusieurs types d'applications (QoS, pare-feu, NAT). Il se compose de deux niveaux, le niveau de protocole de transport, responsable de transporter les messages de signalisation entre les nœuds, et le niveau de protocole de signalisation, spécifique à chaque application de signalisation.

Un autre grand défi fait face à la QoS de bout-en-bout est l'hétérogénéité des réseaux. Les technologies des réseaux varient énormément aussi bien au niveau des réseaux d'accès (xDSL, Ethernet, UMTS, CDMA, Wi-Fi, WiMAX, VSAT …) qu'au niveau des réseaux de transport (MPLS, ATM, Carrier Ethernet …). Chaque technologie est dotée de son propre paradigme QoS et sa manière d'implémentation. Que sera-t-il la QoS de bout-en-bout d'une communication qui traverse plusieurs domaines de technologies différentes ?

Une des premières solutions complète traitant la question d'offrir une qualité de service de bout-en-bout dans un environnement plus général : multi-opérateur, multiservice et multi-technologie est le système EuQoS (End-to-end Quality of Service support over heterogeneous networks). C'est le fruit du projet européen EuQoS. L'idée du projet EuQoS est de concevoir un Framework abstrait, commun et indépendant des technologies des réseaux. L'élément fondamental de ce Framework est les classes de services de bout-en-bout, e2e CoS (end-to-end Class of Service CoSs). Elles sont définies suivant : (i) la nature des applications (real time - RT, non real time - NRT, débit constant - CBR, débit variable – VBR, élastique, etc.) et (ii) les objectifs de QoS (délai de transfert, variation de délai, taux de perte, etc.). Ces classes abstraites ont une portée de bout en bout sur tous les domaines. Le mappage de ces e2e CoS en classes de services spécifiques à la technologie sous-jacente de domaine (telles que les classes AF et EF de DiffServ par exemple) est réalisé au sein de chaque domaine.

Problématique traitée et solution proposée

Le système EuQoS mappe chaque application à la classe de service correspondant à sa nature. Mais il ne peut pas assurer la QoS si la nature de l'application est inconnue ; c'est dans le cas, par exemple, où l'utilisateur veut établir une connexion sécurisée de bout-en-bout satisfaisant certaines paramètres de QoS, mais il ne donne aucune information sur la nature du trafic qu'il va véhiculer.

Pour résoudre ce problème, nous démontrerons d'abord, que l'agrégation d'un flux en une classe conçue initialement pour des applications ayant une nature différente de sa nature entraine un effet néfaste sur les paramètres de QoS même s'ils ont mêmes les objectifs de QoS. Par exemple, l'agrégation d'un flux de VoIP (c.-à-d. de nature CBR) en classe conçue seulement pour les applications

VBR altère au final la QoS offerte par la classe même si les objectifs de QoS demandés par le flux VoIP sont assurés par la classe VBR. Nous prenons comme exemple les classes spécifiées par le projet Aquila, prédécesseur d'EuQoS qui visait de concevoir et d'implémenter d'une architecture de QoS inter-domaine dans un environnement IP.

Les résultats obtenues, et même attendues, nous ont amené à proposer une solution alternative qui consiste à traiter les flux de nature inconnue individuellement. Notre contribution consiste à ajouter au système EuQoS la capacité d'offrir le service par flux (en plus du service par classe déjà fourni) qui réserve de ressources pour chaque flux de bout-en-bout. Nous pensons que ce service est très intéressant aux personnes, et surtout aux entreprises, qui cherchent un service Internet satisfaisant à la fois : les paramètres de QoS et les exigences de la sécurité.

Il est à noter que dans ce contexte, le problème de « scalabilité » (mise à l'échèle), qui peut s'imposer, est minimal car la solution proposée ne concerne qu'une partie particulière du flux. De plus, l'application d'un mécanisme d'agrégation de flux de service garantie permet de minimiser d'avantage le problème de scalabilité.

1 QoS intra-domaine

Avant d'entrée dans la QoS de bout-en-bout, il est nécessaire d'aborder les concepts, les architectures et les technologies définies pour offrir la QoS à l'intérieur d'un domaine. Ce chapitre expose premièrement la définition et les paramètres de QoS (section 1.1) et puis, les deux architectures (IntServ et DiffServ) définies par l'IETF (Internet Engineering Task Force) pour porter la gestion de QoS au réseau Internet. Nous allons aborder les notions clés de ces architectures et les mécanismes de la gestion de trafic et de la gestion de ressources au plan de donnée et au plan de contrôle (sections 1.2 et 1.3). Ensuite nous voyons le réseau MPLS, qui est un paradigme basé sur la commutation de labels posés entre le niveau réseau et le niveau liaison de données, et son couplement avec l'ingénierie de trafic, qui permet d'outrepasser la limite de routage basé sur l'adresse de destination, afin d'obtenir une meilleur optimisation de l'exploitation de ressources (sections 1.4 et 1.5). En fin, nous explorerons le routage à QoS et, en particulier, le problème de trouver le meilleur chemin en respectant plusieurs contraintes. Les dernières sections (sections 1.6 et 1.7) décrit en bref les approches et les algorithmes proposés pour résoudre ce problème.

1.1 Définitions

Nombreuses propositions ont été apportées par divers organismes de standardisation ainsi que par la communauté Internet pour définir la notion de QoS offerte par le réseau. Elles reposent sur plusieurs critères : performance, disponibilité, fiabilité, sécurité, etc. La QoS a été définie par le standard ISO-9000 comme « le degré pour lequel un ensemble de caractéristiques inhérentes satisfont les exigences ». Un autre standard, ISO 8402 [ISO8402], définit la QoS comme « la totalité des caractéristiques d'une entité qui portent sur ses aptitudes à satisfaire les besoins applicatifs fixés ou implicites ». La recommandation ITU-T (International Telecommunication Union – Telecommunication Standardization Sector) X.902 [X902] la définit comme « l'ensemble des qualités liées au comportement collectif d'un ou plusieurs objets ». La recommandation ITU-T E.800 [E800] introduit le concept utilisateur/service par « l'effet collectif de la performance du service qui détermine le degré de satisfaction de l'utilisateur de service ». L'IETF [RFC2386] à son tour l'a défini comme « un ensemble de service pré-requis à remplir par le réseau lors du transport d'un flux ».

Par ailleurs, une grande majorité de la communauté scientifique définit la QoS suivant deux points de vue [Racaru]:

- le point de vue utilisateur : les caractéristiques quantitatives et qualitatives attendues et perçues d'un service ;
- le point de vue fournisseur : la qualité prévue et effectivement fournie.

Les métriques de réseau ont été définies dans le but d'établir une base commune de connaissance au niveau des performances et de la fiabilité du réseau afin d'en obtenir une connaissance précise pour les utilisateurs et pour les fournisseurs de services Internet [RFC2330]. Ces métriques sont définies par la communauté scientifique et les organismes de standardisation au sein de l'IETF et de l'ITU. La métrique réseau est une quantité soigneusement définie exprimant un niveau de performance directement lié au fonctionnement et la fiabilité de ce réseau (Internet par exemple). Cette quantité doit être déterminée de la manière la plus précise possible et doit être exprimée en unités de mesure universelles. Par exemple : la seconde pour la mesure des délais, etc.

Les métriques utilisées pour décrire la QoS [XiPeng] :

Le délai **:** constitué du délai de l'application et du délai du réseau. Le délai de l'application est le temps pris par l'application pour traiter le paquet de données aux bouts de la communication alors que le délai du réseau est le temps entre la mise du premier bit du paquet sur le canal de transmission d'un bout et la réception du dernier bit du paquet de l'autre bout.

Le délai du réseau se compose de trois délais :

Le délai de transmission: C'est la durée nécessaire pour mettre le paquet sur le canal de transmission. Ce délai dépend de la taille de paquet et le débit de canal. Par exemple, pour un paquet de 1500 octet, le délai de transmission sur une liaison DSL de 380 Kbps upstream prend 32 ms, et sur une liaison SDH de 155 Mbps (STM-1) prend 0.08 ms.

Le délai de traitement de paquet : C'est la durée de traitement du paquet dans les différents équipements de réseau (Network Devices). Il comprend le temps de recherche dans la table de routage, le temps de séjour dans la file d'attente, etc.

Le délai de propagation: C'est la durée que prend le signal portant les données sur la distance. Il dépend du type de canal et l'environnement.

L'impact du délai sur la satisfaction de l'utilisateur dépend du service utilisé. Un délai moins de 150 ms est requis pour la voix de haute qualité et moins de 200 ms pour la voix de qualité acceptable.

La gigue : ou la variation de délai, c'est la différence entre le délai de paquet et le délai choisi comme référence (le max ou la moyenne dans un intervalle précis).

Dans un scénario de la téléphonie, si la variation de délai est grande, l'interlocuteur ne peut pas savoir si le silence vient d'un vrai silence de son correspondant ou d'un retard causé par le réseau. Alors que, si la variation est très petite, il peut bien s'adapter à la situation et estimer l'état de son correspondant. Généralement une variation de délai moins 50 ms est très acceptée pour un service de la voix ou de la vidéo.

La perte de paquets : C'est le pourcentage de paquets perdus de bout en bout. Elle est due à l'application d'un mécanisme de prévention ou de guérison de congestion, à la panne d'un équipement réseau, à la vieillesse de paquet (le champ TTL de l'entête d'un paquet IP).

Certaines applications, comme les applications multimédia, peuvent tolérer jusqu'à certain seuil de perte sans altérer la perception de l'utilisateur du service.

Le débit : C'est le nombre maximal de bits que l'application peut envoyer par unité de temps. L'influence du débit sur la perception de l'utilisateur se voit, par exemple, dans le HDTV qui exigent un minimum de débit pour fonctionner, et dans le cas où un débit élevé un transfert de grand fichier prend moins de temps [XiPeng].

L'influence de ces facteurs sur la perception de l'utilisateur varie d'une application à une autre. C'est au réseau donc de gérer ses ressources de manière à répondre aux exigences des services offerts et à minimiser l'effet de ces facteurs sur les services. Dans le monde Internet plusieurs architectures et mécanismes ont apparus pour le rendre à la hauteur de ce nouveau défi. Ils adressent deux grands axes : allocation de ressources et optimisation de performances.

Sous allocation de ressource on distingue les problèmes de paquets retardés ou supprimés à cause de l'insuffisance de ressources. Un réseau, dans sa forme simple, est vu comme un ensemble limité de ressources partagées entre les utilisateurs et chacun peut injecter tant de paquets au plus vite que

possible sans aucun contrôle. Au sein de réseau, les paquets de différentes applications sont servis à l'ordre d'arrivée (FIFO). En cas de congestion, les paquets seraient détruits sans différenciation. C'est le service *Best-Effort*.

Le réseau qui supporte la QoS est un réseau doté d'un processus d'allocation de ressources qui décide d'attribuer quelle ressource à quel service. Deux architectures ont été standardisées pour répondre à cette question dans le monde Internet : Intégration de service (IntServ) et Différenciation de service (DiffServ). Ces deux architectures ont introduit de nouveaux concepts et primitives :

- Framework d'allocation de ressources qui permet de la garantie de ressources et la différenciation de services,
- nouveaux modèles de services en plus de model *Best-Effot*,
- langage de description des ressources requises et des ressources garanties,
- mécanismes effectuent l'allocation de ressources.

L'optimisation de performances traite la question de l'organisation de ressources du réseau de manière très efficace de telle sorte qu'on répond à autant de demandes de QoS avec le moindre du coût.

1.2 Intégration de services (Integrated Services)

L'intégration de services était le premier pas vers la dotation de l'internet de la capacité de supporter la QoS. Deux motifs importants poussaient vers cette orientation [Wang]:

- les expérimentations réalisées sur le réseau MBONE, réseau expérimental pour la transmission de trafics multicast sur Internet, qui ont montré la nécessité d'une amélioration significative de l'architecture d'Internet afin de supporter les applications temps réel.
- Le progrès enregistré dans les travaux de recherches concernant le développement des algorithmes et des mécanismes de gestion de ressources réseaux, et particulièrement les travaux sur l'ordonnancement qui ont montré la possibilité de garantir, dans un ordonnanceur à plusieurs niveaux, un délai et une bande passante à un niveau particulier.

L'IETF a établi plusieurs groupes de travail ayant pour vocation l'arrivée à un standard de modèles de services et des protocoles pour Integrated Services. Les principaux groupes étaient:

- Integrated Services (*INTSERV*) working group: responsable de définir l'architecture, les modèles de services, la spécification de flux et le cadre (Framework) des composants de l'intégration de services, à savoir: contrôle d'admission, identification de flux et l'ordonnancement.
- The Resource Reservation Setup Protocol (*RSVP*) working group: responsable de définir le protocole RSVP qui sert à réserver les ressources nécessaire au flux.

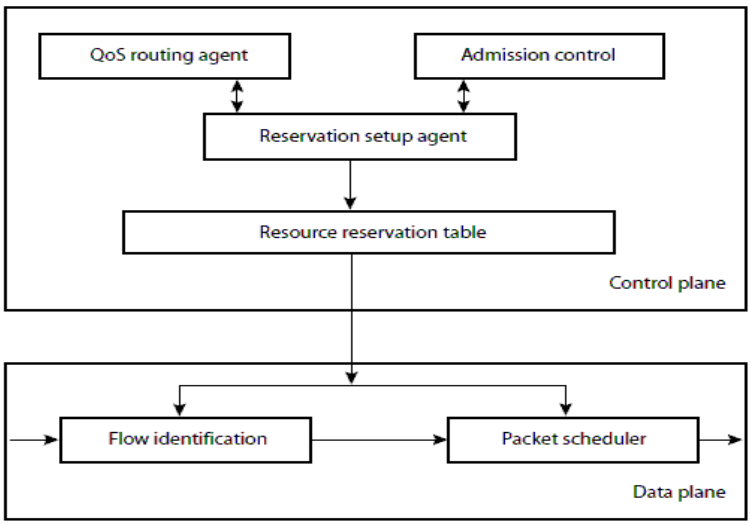


Figure 1.1 : modèle de référence d'Intégration Services.

Comme montre la figure 1-1, IntServ est composé logiquement de deux plans : le plan de contrôle où s'effectue la réservation de ressources, et le plan de données où s'effectue l'acheminement (Forwarding) de paquets en se basant sur l'état actuel de réservation.

Avant de commencer l'envoie de flux, l'émetteur décrit les spécifications de flux et les ressources nécessaires et envoie sa demande de réservation au réseau. La demande passe d'un routeur à un autre pour arriver à la fin au destinataire. Le destinataire retourne la réponse sur le même chemin mais cette fois chaque routeur applique un contrôle d'admission et réserve les ressources nécessaires s'il y en a suffisamment. Le routeur garde les informations sur la réservation de flux dans la table de réservation de ressources. Une fois que la réservation est établie, l'émetteur peut commencer à l'envoie des paquets du flux. A la réception d'un paquet, l'identificateur de flux vérifie à quel flux il appartient en exploitant les informations gardées dans la table de réservation de ressources et le met à la file d'attente propre à son flux. L'ordonnanceur de paquets (Packet Scheduler), un composant important de l'architecture IntServ, est responsable de la sélection du paquet à sortir parmi les paquets séjournant dans les différents niveaux des files d'attente [Wang].

1.2.1 Classes de services d'IntServ

IntServ définit deux classes de services:

- *Guaranteed Service (GS)*: la classe à service garanti fournit une garantie stricte sur le délai maximum de chaque paquet et assure un taux de perte nul (le délai maximum est calculé à partir des informations sur le flux et le débit de service requis).
- *Controlled Load(CL)*: la classe à charge contrôlée fournit le même service qu'un réseau best-effort en situation non congestionnée.

Le but de la définition de la classe CL est d'émuler un service dans un réseau mois chargé, c'est-à-dire un réseau moins congestionné, et par conséquence un peu de perte de paquets. Ce type de service est très adéquat pour les applications qui tolèrent un petit taux de perte sans se soucier de revendiquer la classe GS. Cela est permis grâce aux mécanismes d'ordonnancement tel que WFQ qui permet d'isoler (protéger) les trafics CL des autres trafics GS, et aussi grâce au contrôle d'admission qui limite la quantité de trafic CL acceptés afin de garder un niveau raisonnable de la charge [Peterson].

1.2.2 Spécification de flux

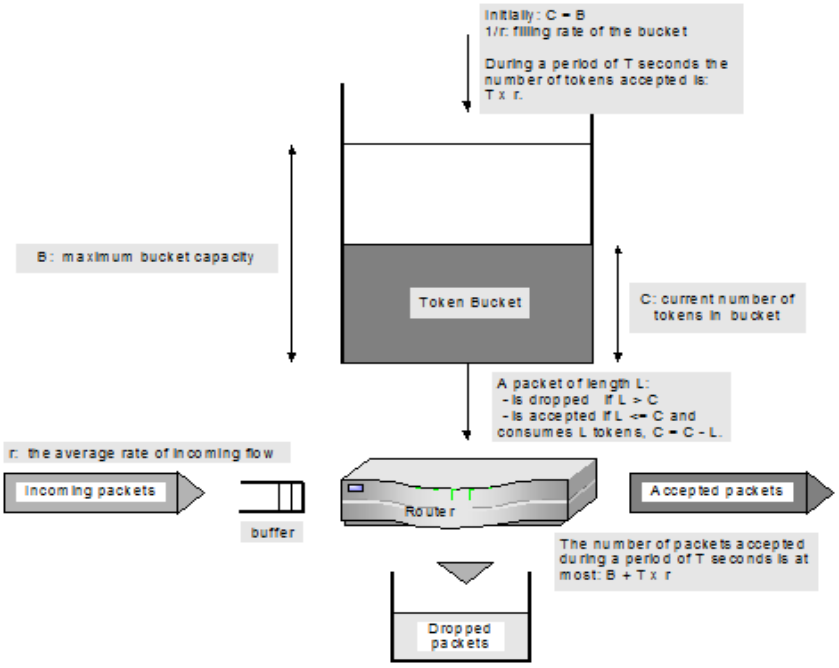


Figure 1.2: le mécanisme de contrôle Token Bucket.

La spécification de flux se compose de deux parties: spécification de trafic *TSpec* qui décrit les caractéristiques de trafic à envoyer, et *RSpec* qui décrit le service demandé.

IntServ caractérise le trafic selon le modèle seau à jetons (Token Bucket) qui constitue d'un seau de capacité *b* se remplissant à un débit de *r* jeton par unité de temps.

TSpec décrit le trafic par les paramètres suivants [Wang]:

- *Bucket rate (r)* (bytes/second): le débit de l'arrivée de jeton.
- *Peak rate (p)* (bytes/second): le débit maximal de transfert.
- *Bucket depth (b*) (bytes):La capacité du seau.
- *Minimum packet size (m)* (bytes): la taille minimale de paquet.
- *Maximum packet size (M)* (bytes): la taille maximale de paquet ;

alors que RSpec est spécifique à la classe GS et décrit le service requis par le flot en termes de [Wang]:

- *Service rate (R)* (bytes/second): le débit de service requis.
- *Slack Term (S)* (microseconds): ce paramètre est propre au protocole RSVP. Il représente le délai à compenser (crédit) en cas où un nœud ne peut pas satisfaire le délai nécessaire pour garantir le délai de bout en bout.

A partir de ces informations on peut calculer le délai maximal de bout en bout de séjour dans les files d'attente de différents nœuds au long de chemin. Pour calculer le délai maximal total de bout en bout on lui ajoute deux autres délais :

- C_{tot} : total des délais pris par le paquet pour passer d'un nœud à un autre. Il dépend du débit et la taille de paquet.
- D_{tot} : total des délais de traitement de paquet au sein des nœuds au long du chemin.

Le délai maximal de bout en bout s'écrit [Chao] :

$$\frac{(b-M)(p-R)}{R(p-r)} + \frac{M+C_{tot}}{R} + D_{tot} \quad (p > R \geq r)$$

$$\frac{M+C_{tot}}{R} + D_{tot} \quad (R \geq p \geq r)$$

1.2.3 Contrôle d'admission

Quand un nouveau flux demande une garantie de services, le contrôle d'admission vérifie s'il est possible d'accepter ce flux défini par ses spécifications TSpec et RSpec sans influer sur les garanties de services déjà attribués aux autres flux. Pour la classe GS l'agent de contrôle d'admission est responsable de prendre une définitive décision, oui ou non, à la demande en basant sur les paramètres de trafic (parameter-based approach). Alors que pour la classe CL la décision se base sur les mesures de la charge de réseau, c'est-à-dire de façon probabiliste (measurement-based approach) [Wang].

1.2.4 Classification et Ordonnancement de paquets

Une fois que le flux et le service requis sont spécifiés et la réservation est établit au long du chemin de la source au destinataire, Il reste à délivrer les services requis aux flux de manière à :

- associer chaque paquet à la réservation appropriée à son flux (classification) ;
- gérer les paquets dans les files d'attente de manière qu'ils reçoivent leurs propres services (Scheduling).

La classification s'effectue en examinant les cinq champs : adresse source, adresse destination, numéro de protocole, port de la source, port de la destination (il est possible d'utiliser le champ FlowLabel de l'entête IPv6).

La tâche centrale de l'ordonnanceur de paquets consiste à sélectionner le paquet à transmettre lorsque le lien sortant est prêt. Il affecte directement la bande passante et le délai de paquets. WFQ (Weighted Fair Queuing) est une classe d'algorithme d'ordonnancement de paquets qui supportent l'allocation de bande passante et les limites de délai. Comme montre la figure 1.3, la bande passante allouée à chaque file est égale à : $r_i = R * w_i / (w_1 + w_2 + ... + w_n)$.

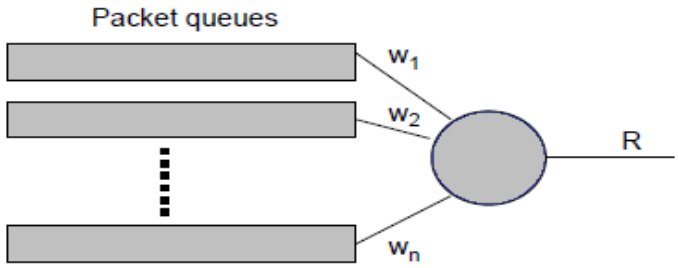


Figure 1.3: le mécanisme de contrôle WFQ.

1.2.5 Le protocole de signalisation RSVP

RSVP est un protocole de signalisation pour allouer dynamiquement de la bande passante aux applications orientées connexion dans un environnement traditionnellement datagramme. RSVP est utilisé dans le modèle IntServ, mais il peut aussi être utilisé hors de ce contexte (par exemple pour établir des chemins MPLS).

RSVP rend obligatoire la demande de QoS par le récepteur (l'application participante) plutôt que par l'émetteur (l'application source). Le récepteur apprend les spécifications du flux et demande les réservations qui lui sont nécessaires. Cela est très utile dans le cas d'une transmission multicast. Il a aussi la possibilité de partager les réservations entre plusieurs flux (émetteurs) de la même session selon différentes modalités (utile par exemple dans le cas de l'audioconférence où un seul des participants parle et donc émet à tout instant).

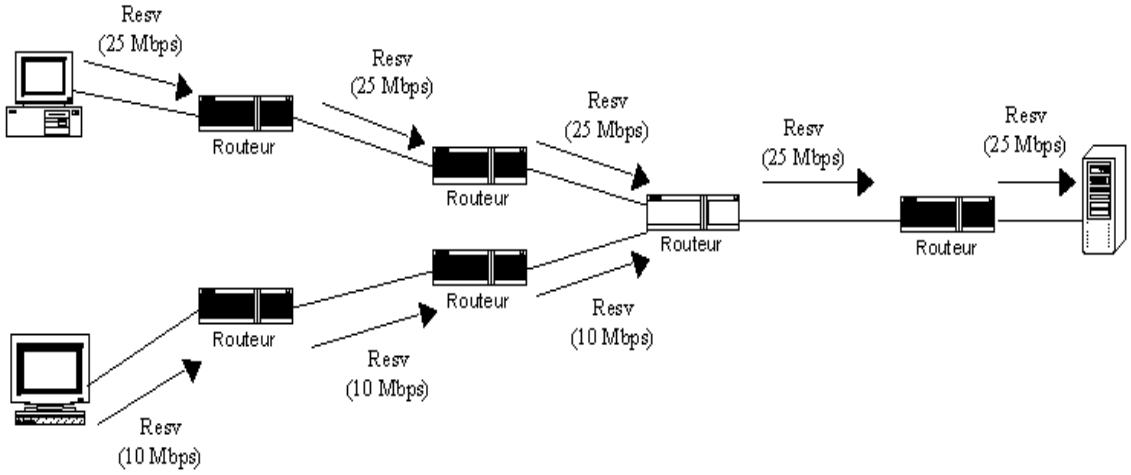


Figure 1.4: la signalisation par RSVP.

1.2.5.1 Les messages de RSVP

Sept types de Messages RSVP ont été prévus [Wang]:

- *PATH*: envoyé par la source pour indiquer la liste des routeurs du chemin suivi par les données;
- *RESV*: demande de réservation;
- *Patherr*: message d'erreur concernant le chemin;
- *ResvErr*: message d'erreur de demande de réservation;
- *PathTear*: indique aux routeurs d'annuler les états concernant la route;
- *ResvTear*: indique aux routeurs d'annuler les états de réservation (fin de session);
- *ResvConf* (optionnel): message de confirmation envoyé par le routeur au demandeur.

1.2.5.2 Fonctionnement de RSVP

La session RSVP est identifiée par l'adresse IP de destination (éventuellement multicast) et un numéro de session. Plusieurs flux peuvent exister dans la même session, un par émetteur.

L'application qui désire participer dans cette session en tant qu'émetteur envoie un message PATH contenant le TSpec de flux qui est routés jusqu'au(x) destinataire(s). Dans chacun des routeurs qu'il traverse, le message PATH crée un *Path State* contenant essentiellement l'adresse du routeur précédent (*Previous Hop*) ; créant ainsi un *arbre de parcours inverse*. Chaque routeur ajoute également au message PATH des informations, contenues dans l'objet ADSPEC (*advertising specification*), sur les qualités de service qu'il est capable de fournir, ce qui correspond à un mécanisme OPWA (*one pass with advertising*). Un appel à la fonction de routage permet ensuite de déterminer le lien de sortie à utiliser pour envoyer le message vers le nœud suivant.

Quand l'application destinataire reçoit le message PATH, elle connaît alors les caractéristiques du flux de données proposé et celles du chemin parcouru. Elle spécifie la qualité de service qu'elle désire obtenir en envoyant un message RESV qui retournera vers l'émetteur grâce aux Path States, déjà installés par le message PATH. Ce message RESV contient le " TSpec du récepteur " caractérisant

le trafic pour lequel le récepteur veut réserver de ressources et le RSpec (*resource specification*) spécifiant les ressources à réserver. À chaque nœud, un contrôle d'admission est réalisé pour vérifier si suffisamment de ressources sont disponibles pour fournir le service demandé. Si c'est le cas, un *Resv State* est créé, contenant les informations relatives à la réservation à effectuer et le message est envoyé au nœud amont. Les *Resv States* créés par différents récepteurs (de la même session) et associés au même lien de sortie sont accumulés dans un *Traffic Control State* contenant la réservation globale sur ce lien pour cette session. Les états de classification et d'ordonnancement peuvent alors être mis en place [Deleuze].

Tous ces états sont de type *soft states*, ce qui implique que les messages sont réémis périodiquement pour les rafraîchir. Cela permet également, en association avec les mécanismes d'IP multicast, à de nouveaux participants de se joindre à la session dynamiquement. De plus, les états " abandonnés " ne resteront pas indéfiniment dans le réseau. L'effacement des états se fait normalement de manière explicite avec les messages PTEAR (*path tear*) et RTEAR (*resv tear*) mais l'effacement d'un *soft state* à l'expiration d'un temporisateur de survie (*cleanup timeout*) permet d'assurer la robustesse du protocole sans nécessiter la fiabilité de la transmission des messages.

1.2.6 Limites de IntServ

L'écueil principal qui entrave le déploiement de l'architecture Intégration de Services est les difficultés liées au passage à l'échelle (*scalibilité*).

- En effet le paradigme qu'il utilise, la gestion des ressources par flux, implique la maintenir dans chaque routeur un triple état pour chaque flux traversant ce routeur (indépendamment d'autres états globaux qui ne posent pas de problème d'échelle). Ces états sont de:
  - *signalisation* : état de contrôle (par exemple les *resv* et *path states* de RSVP) ;
  - *classification* **:** pour l'identification des paquets d'un flux ;
  - *ordonnancement* **:** pour l'allocation des ressources réservées ;

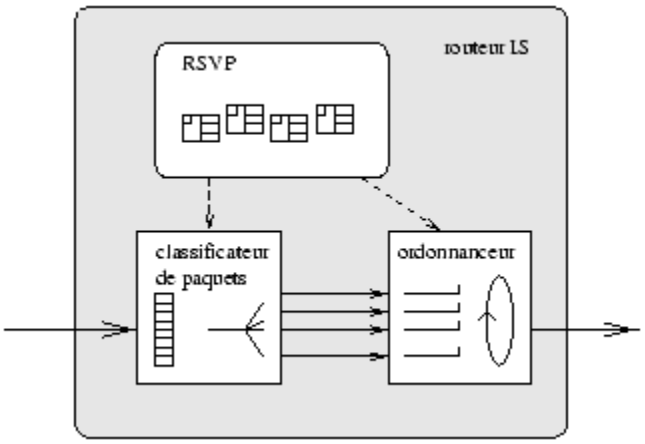


Figure 1.5: le problème de scalabilité dans l'IntServ.

- La maintenir des états de réservations de tous les flux courants dans les routeurs nécessite l'échange d'un grand nombre de messages de rafraichissement. Ce qui augmente la charge du réseau et, par conséquence, augmente la probabilité de supprimer les paquets des trafics non garanties y compris les messages de rafraichissement eux-mêmes et conduit ainsi à la libération de ressources réservées.

1.3 Différentiation de services

En réaction aux limites et aux difficultés de déploiement du modèle IntServ, et face aux demandes de plus en plus pressantes des fournisseurs d'accès qui veulent pouvoir proposer différents types de services, un nouveau groupe de travail de l'IETF, The Differentiated Services Working Group ou DiffServ, a été chargé d'étudier une nouvelle approche, appelée la différenciation de services. Le groupe DiffServ a proposé d'abandonner le traitement des trafics sous forme de flux pour les traiter sous forme de classes. Chaque classe est identifiée par une valeur codée dans l'en-tête IP. La classification de flux est opérée à l'entrée du réseau (ou de la zone où la différenciation de service est mise en place - *domaine DS*) par les nœuds de bordure (boundary nodes) qui gèrent les états par flux [Lochin].

Un domaine DS est une zone administrative fournissant la différenciation de services, c'est à dire fonctionnant avec un ensemble commun de politiques de provisionnement de réseau et de définitions de PHBs. Une région DS est un ensemble contigu de domaines DS qui peuvent offrir la différenciation de services sur des chemins traversant ces domaines.

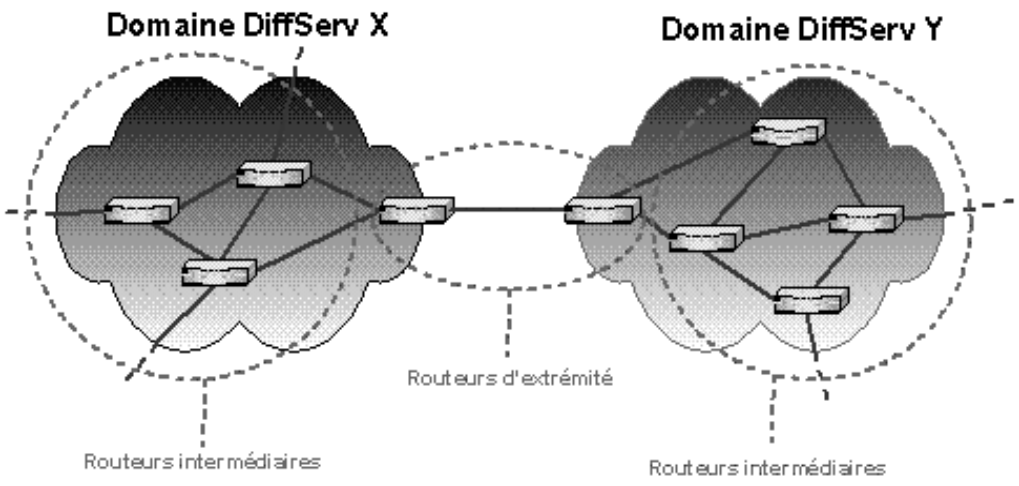


Figure 1.6: domaines DiffServ interconnectés.

Le contrôle d'admission est fait à priori par la définition d'un contrat pour chaque classe de trafic et par le dimensionnement des ressources pour pouvoir garantir ce contrat. La complexité du traitement est alors concentrée dans les routeurs aux frontières de réseau. Ils effectuent les opérations "complexes" de contrôle de la validité du contrat pour les différentes classes de trafic. Dans le cœur de réseau, le traitement est plus simple, ce qui autorise une circulation plus rapide des données.

Les routeurs DiffServ traitent les paquets en fonction du type de classe codé dans l'entête IP (champ DS) selon un comportement spécifique : le PHB (*Per Hop Behaviour*). Chaque ensemble de paquets appartenant à une classe reçoit alors le même traitement et chaque classe est codée par un DSCP (*DiffServ Code Point*). Un PHB est défini également par la priorité qu'il a sur les ressources par rapport à d'autres PHB.

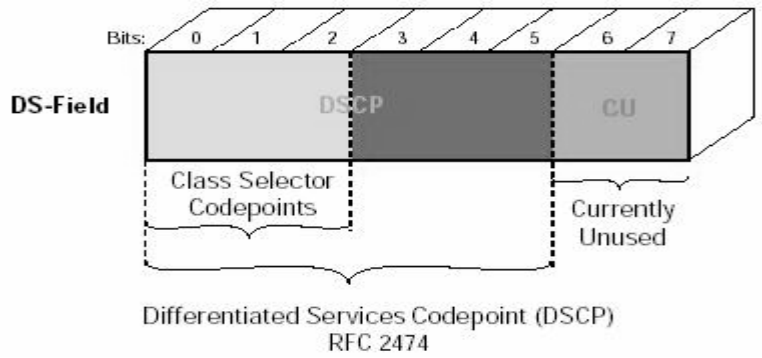


Figure 1.7: le champ DSCP de DiffServ.

DiffServ définit la sémantique générale des PHBs et non les mécanismes spécifiques qui permettent de les implémenter. Les PHBs sont définis une fois pour toutes, tandis que les mécanismes peuvent être modifiés et améliorés, voire être différents, suivant le type de réseau sous-jacent [Pujolle].

1.3.1 Les classes de services

Outre le service BE (best-effort), deux PHB sont définis dans DiffServ:

- EF (Expedited Forwarding) ou Premium Service (défini dans [RFC2598]) : correspond à la priorité maximale et a pour but de garantir une bande passante avec de taux de perte, de délai et de gigue faible en réalisant le transfert de flux à fortes contraintes temporelles comme la voix ou la visioconférence;
- AF (Assured Forwarding), (défini dans [RFC2597]) : regroupe plusieurs PHB garantissant l'acheminement de paquets IP avec une haute probabilité et sans tenir compte le délai ; cette famille de PHB est scindée en 4 classes garantissant de fournir une bande passante et un délai minimal, chaque classe comprenant 3 niveaux de priorité (*Drop Precedence*) ;

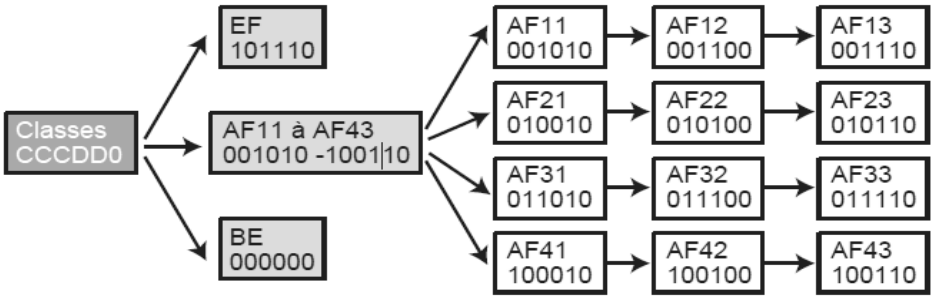


Figure 1.8: classes et sous classes de DiffServ.

1.3.1.1 EF (Expedited Forwarding)

La classe Expedited Forwarding correspond à la valeur 101110 de DSCP et a pour objectif de fournir un service de transfert équivalent à une ligne virtuelle dédiée à travers le réseau de l'opérateur. Le contrat porte sur un débit constant. Les paquets excédentaires sont lissés ou rejetés à l'entrée pour toujours rester conforme au contrat. L'opérateur s'engage à traiter ce trafic prioritairement. De plus, les flux ne doivent avoir que très peu de perte, et la gigue doit être minimale et la bande passante garantie. Ils sont dotés d'une priorité forte dans les nœuds mais doivent être contrôlés pour que la somme des trafics provenant des différentes sources et passant sur la même liaison ne dépasse pas la capacité nominale déterminée par l'opérateur.

Plusieurs solutions permettent de réserver la bande passante proposée aux flux de paquets EF. Un protocole de type RSVP, par exemple, peut effectuer les réservations de bande passante nécessaires. Une autre solution consiste à utiliser un serveur spécialisé dans la distribution de la bande passante, ou Bandwidth Broker (voir le chapitre suivant). Ce serveur de bande passante réalise le contrôle d'admission en proposant une réservation centralisée [Pujolle].

1.3.1.2 AF (Assured Forwarding)

Il s'agit en fait d'une famille de PHB (*PHB group*). Quatre classes de "traitement assuré'' sont définies. Chacune comporte de trois niveaux de priorité (*drop precedence*) suivant que l'utilisateur : respecte son contrat, le dépasse légèrement ou est largement en dehors. Les classes sont donc choisies par l'utilisateur et restent les mêmes tout au long du trajet dans le réseau. Tous les paquets d'un flux appartiennent à la même classe. A l'intérieur de chaque classe, un algorithme de rejet sélectif différencie entre 3 niveaux de priorité. En cas de congestion dans une classe AF, les paquets de basse priorité sont détruits en premier. La priorité peut être modifiée dans le réseau par les opérateurs en fonction du respect ou non des contrats.

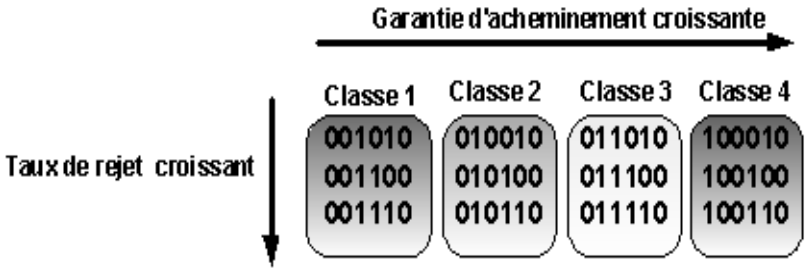


Figure 1.9: priorité d'acheminement et de rejet.

Un domaine implémentant des services AF doit être, par l'intermédiaire des routeurs de frontière, capable de contrôler les entrées de trafics AF pour que la qualité de service déterminée pour chaque classe AF soit satisfaite. Pour cela, les routeurs de frontière doivent mettre en place des mécanismes de mise en forme de trafic (shaper), de destruction de paquets (dropper), d'augmentation ou de diminution des pertes de paquets par classe AF et de réassignation de trafics AF dans d'autres classes AF moins prioritaire.

1.3.2 Les routeurs de bordure

Les traitements effectués dans les routeurs de bordure sont généralement les plus complexes et correspondent à un contrat (*SLA* : Service Level Agreement) préalablement établi entre l'utilisateur et l'opérateur de réseau.

Le SLA définit le type de service offert ainsi qu'un ensemble de règles pour le conditionnement du trafic :

- taux de disponibilité moyen, taux de perte moyen.
- délai borné, délai moyen, gigue moyenne.
- Les types de micro-flux faisant partie de chaque classe.
- La valeur de marquage DSCP.
- La bande passante allouée, pics de données acceptés.
- Le choix d'une politique à appliquer (*Policing*) en cas de dépassement du contrat parmi les possibilités suivantes:
  - Transmission.

- Rejet de paquets.
- Abaissement du niveau de priorité (changement de classe).
- Lissage des flux (*Chaping*).

- La taille des buffers de files d'attente.

Les routeurs de bordure ont également un rôle particulier dans l'interfaçage entre les PHB de chacun des deux domaines qu'ils associent. Ils pourront éventuellement effectuer un reconditionnement de paquets si les règles de classification et de conditionnement ne sont pas les mêmes sur le domaine de destination. Un contrat SLA d'une organisation X peut par exemple considérer un flux "Gold" (classe de service AF 3) d'un client Y entrant comme un flux "Silver" (classe AF 2) au sein de son domaine. Dans ce cas, le routeur de bordure du domaine DiffServ du client Y devra alors effectuer des opérations de conditionnement de trafic pour le trafic sortant.

Les traitements qu'effectuent les routeurs de bordure sont :

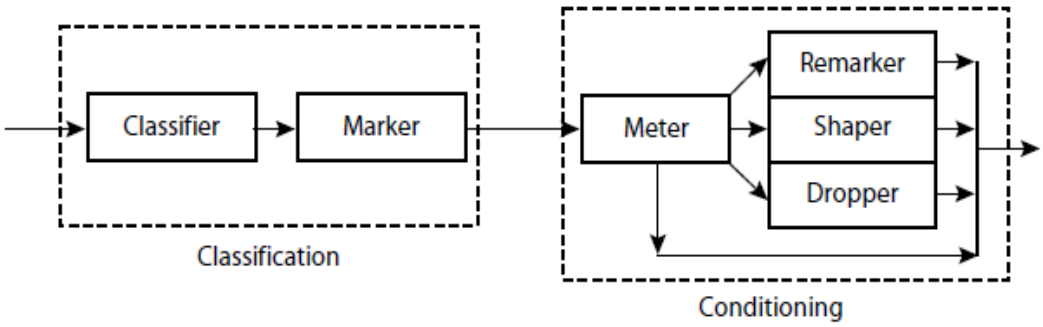


Figure 1.10: les éléments des routeurs de bordure.

1.3.2.1 Classification

La classification s'effectue suivant une ou plusieurs valeurs contenues dans l'entête de paquet IP (exemple : adresse source - destination, port source - destination, protocol ID, ...). Une fois le paquet est identifié, il est dirigé vers la fonction de marquage appropriée.

Le groupe de travail DiffServ ne précise pas comment le classificateur est paramétré pour effectuer La classification ou plus exactement, qui le paramètre. Cela soit faite manuellement, aux bons soins de l'administrateur qui paramètre les tables de marquage des paquets en fonction d'une table d'adresse source, par exemple, donnée au routeur de bordure, soit par le biais d'un protocole de signalisation; RSVP pourrait d'ailleurs très bien faire l'affaire. En effet, celui-ci n'est pas un protocole de signalisation propre à IntServ uniquement, et on pourrait l'utiliser afin de signaler les classes à traiter par les routeurs.

1.3.2.2 Conditionnement

Une fois un paquet de flux est identifié et marqué par le classificateur, il le dirige vers le module de conditionnement spécifié pour continuer le processus de traitement. Le conditionneur de trafic contient un ensemble d'éléments tels que le " Meter ", le " Shaper " et le " Dropper ". Le travail de Meter consiste à vérifier que les flux entrants ne dépassent pas le contrat (SLA) configuré dans le routeur. Cette information est transmise aux modules de remarquage et de shaping/dropping qui effectueront alors des actions conformes à la politique fixé dans le SLA. Le Shaping peut s'effectuer lorsque les flux d'une classe outrepassent le contrat SLA prédéfini. Cette fonctionnalité n'est pas systématique et correspond elle aussi à une règle fixé dans le SLA. Les paquets sont alors mis en file

d'attente afin d'être transmis un peu plus tard lorsque le débit de la classe sera considéré comme étant dans le profil du contrat.

Le rejet des paquets intervient pour garantir le débit fixé pour chaque classe de service. Dans le cadre de lissage, et comme les files d'attente ont une taille finie, les dépassements trop importants de profil peuvent aussi provoquer un rejet des paquets.

1.3.3 Les routeurs intermédiaires

Le problème principal de l'architecture IntServ était sa complexité de mise en œuvre dans les équipements intermédiaires via le mécanisme de réservation de ressources RSVP. Le modèle DiffServ procède donc différemment en définissant des traitements plus simples dans les routeurs intermédiaires. A l'arrivée d'un paquet, il travaille à l'aiguiller vers un port de sortie et à déterminer son nouveau PHB afin de les traiter au niveau d'ordonnancement suivant sa classe. Plusieurs politiques d'ordonnancement peuvent être utilisées : WFQ, WRR ou priorités fixes. Les opérateurs s'orientent vers une combinaison de ces politiques avec une priorité fixe pour le trafic temps réel et un ordonnancement WFQ pour les autres classes. C'est le champ DSCP des paquets qui permet d'affecter les paquets à une file d'ordonnancement particulière [Rachdi].

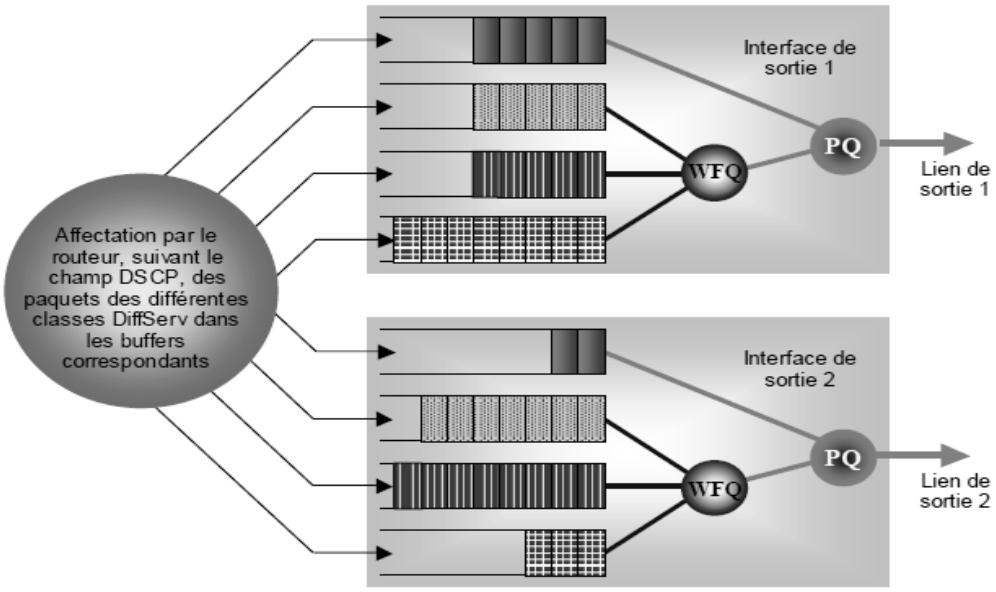


Figure 1.11: l'ordonnancement dans les routeurs DiffServ.

1.3.4 Limites de DiffServ

DiffServ nécessite en effet d'établir préalablement un contrat dans tous les équipements de domaine. Ceci implique une connaissance approfondie des applicatifs qui peuvent transiter sur le réseau ; et ce peut se révéler parfois difficile à appliquer et moins flexible par apport à l'architecture IntServ. De plus le conditionnement de trafic de manière très conservatrice conduit à une forte inutilisation de ressources de réseau.

En plus, le paradigme Datagramme ne permet pas d'optimiser les performances de réseau [Zheng]. Les protocoles de routage d'Internet orientent tous les paquets ayant même préfix d'adresses de destination vers la même route. Le routage qui base sur l'adresse de destination (destination-based routing) ne permet pas de partager les trafics de même destination sur plusieurs routes, ce qui conduit à une mauvaise exploitation de ressources. En plus, la décision de routage est prise localement, c'est-à-dire le routeur choisit le meilleur chemin selon son avis. Si tous les routeurs choisissent par exemple

le plus court chemin, ce chemin devient congestionné. MPLS TE, la gestion de réseau par politique (PBNM) et le routage à QoS sont des solutions proposées pour combler cette lacune et permettre de tenir compte d'autres facteurs dans le choix de route.

1.4 MPLS (MultiProtocol Label-Switching)

MPLS est une norme proposée par l'IETF pour l'ensemble des architectures et des protocoles de haut niveau (IP, IPX, AppleTalk, etc.). Le protocole MPLS, basé sur le paradigme de changement de label, s'inspire directement de l'expérience acquise de l'ATM (étiquettes VPI/VCI). Ce mécanisme est aussi similaire à celui de Frame Relay et de liaisons PPP.

L'idée de MPLS consiste à rajouter un label de couche 2 aux paquets IP dans les routeurs d'entrée au réseau (LER : Label Edge Routers) entre la couche IP et la couche de liaison de données, et à acheminer les paquets en basant seulement sur le contenu de label. Le réseau MPLS crée des chemins LSP (Label Switching Path), similaires aux PVC d'ATM. Ces LSP définissent une route de bout en bout par la concaténation de labels. A la sortie du réseau MPLS le label est retiré par l'autre routeur de bordure, ce qui permet de retenir le paquet IP original. Les routeurs du cœur de réseau, appelés LSR (Label Switching Routers), vont router les paquets de proche en proche par commutation de labels, oubliant ainsi les adresses IP [Rachdi].

L'entête MPLS se situe entre les entêtes des couches 2 et 3, où l'entête de la couche 2 est celle du protocole de liaison et celle de la couche 3 est l'entête IP. L'entête est composé de quatre champs:

- Le champ Label (20 bits),
- Le champ Exp ou CoS (3 bits) pour la classe de service (Class of Service),
- Un bit Stack pour supporter un label hiérarchique (empilement de labels),
- Et un champ TTL (Time To Live) pour limiter la durée de vie de paquet (8 bits). Ce champ TTL est le même que celui de l'IP.

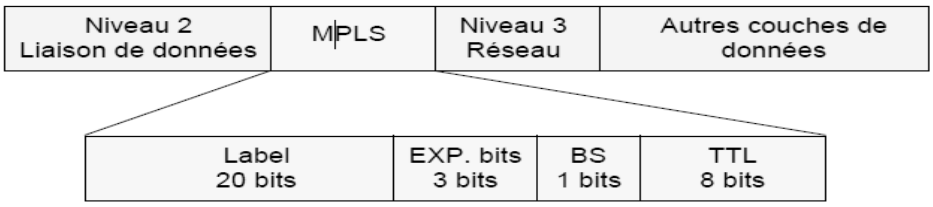


Figure 1.12: entête MPLS.

Le service MPLS est particulièrement adaptable à un service différencié de type DiffServ. En effet, dans son concept, on retrouve la notion de routeurs de frontières et de routeurs de cœur de réseau (edge/core router) ainsi que la notion d'agrégation de flux.

1.4.1 MPLS et l'ingénierie de trafic (MPLS TE)

Le MPLS TE introduit, la possibilité d'orienter les paquets IP vers un chemin qui n'est pas forcément le plus court chemin de niveau 3 (issu du protocole de routage interne du réseau, i.e. RIP, OSPF, IS-IS, EIGRP, etc.), ceci afin de mieux gérer les ressources.

Le MPLS TE, permet donc de sélectionner les routes (ou bien les LSP) en tenant compte la disponibilité de ressources et la charge courante de réseau et la charge attendu. La classe de service et la QoS

requise pour les données peuvent aussi être prises en compte. Le MPLS TE peut être réalisé manuellement ou suivant un processus automatisé, réagissant aux informations relatives à l'état du réseau (charges des liens, pannes, nouveaux trafics).

Avec TE, la meilleure exploitation de ressources devient un problème mathématique d'optimisation dont le but est de trouver la meilleure de toutes les solutions possibles. En d'autre terme, étant donné la topologie fixe du réseau et la matrice fixe de demandes de trafic (source-destination), le problème d'optimisation est alors de trouver les routes de flux qui rendent l'exploitation de réseau plus efficaces [Evans].

Le TE-MPLS est donc une seconde valeur ajoutée à la solution MPLS. Il permet en effet à l'opérateur de faire une gestion plus efficace de son réseau grâce à la possibilité de tailler les « LSP » en fonction des classes de services (CoS) et des FEC (Forwarding Equivalent Class).

Alors que les protocoles à état de lien utilisent l'algorithme SPF (Shortest Path First) pour déterminer le plus court chemin entre lui et tous les autres routeurs du réseau (construction de la table de routage), certaines implémentations de MPLS TE utilisent l'algorithme CSPF (Constrained Shortest Path First) qui consiste à éliminer les liens ne satisfaisant pas les contraintes puis à appliquer l'algorithme SPF.

L'administrateur a la possibilité d'imposer manuellement un chemin explicite à une FEC donnée.

La création d'un LSP TE consiste à [Alvarez] :

- diffuser les informations de routage : en plus des informations de routage diffusées naturellement par les protocoles à état de lien OSPF et IS-IS, MPLS TE introduit d'autres attributs à diffuser : la bande passante disponible, le groupe administratif (flags) et la métrique TE. Au lieu de définir un nouveau protocole pour diffuser ces informations de routage, une modification a été portée sur OSPF et IS-IS pour devenir OSPF-TE, IS-IS-TE respectivement [Braun].
- calculer le MPLS TE LSP: Mise en œuvre de l'algorithme CSPF ou intervention manuelle de l'administrateur pour imposer une route explicitement.
- établir le MPLS TE LSP: Mise en œuvre d'un protocole de distribution de label Request-based (RSVP TE, CR-LDP).

L'exemple de la figure ci-dessous montre des liens caractérisés seulement par trois paramètres: le coût, la bande passante disponible et le groupe administratif [Braun].

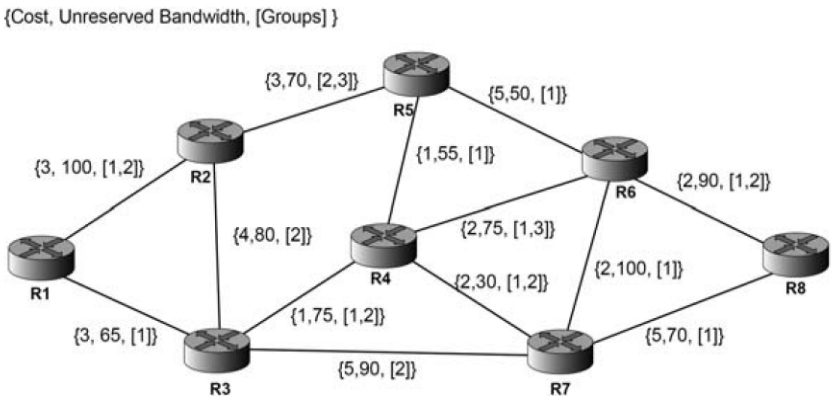


Figure 1.13: exemple de réseau à trois métriques.

On veut calculer la route de R1 à R8 avec la contrainte de ne passer que par les liens ayant une bande passante supérieure à 60 Mbit/s et n'appartenant pas au groupe 3.

Premièrement on élimine les liens ne satisfaisant pas les deux contraintes. Le graphe devient comme ci-dessous:

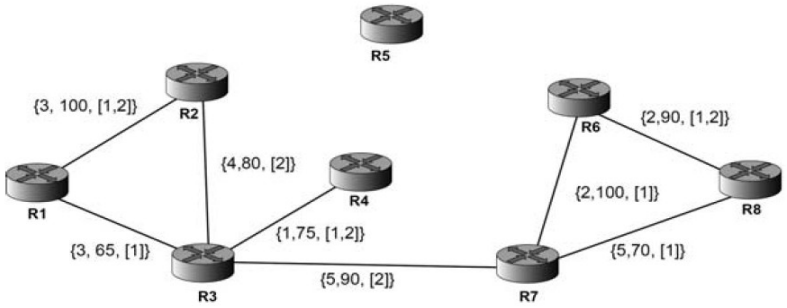


Figure 1.14: élimination de liens insatisfaisante.

En y appliquant l'algorithme SPF en partant du nœud R1, le résultat est le graphe ci-dessous:

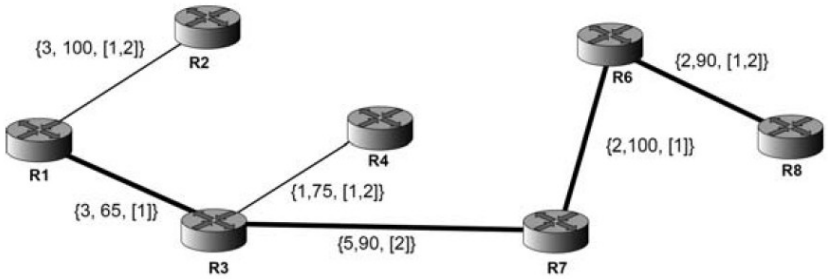


Figure 1.15: graphe de plus court chemin.

Ainsi, le plus court chemin de R1 à R8 respectant les contraintes précédentes est: *R1 → R3 → R7→ R6 → R8* avec un coût de 12.

Il est à noter que sans contrainte, la route aurait été: *R1 → R3 → R4→ R6 → R8* avec un coût de 8.

1.4.1.1 Le protocole CR-LDP

CR-LDP est une version étendue de LDP, où CR correspond à la notion de routage basé sur les contraintes des LSP. Tout comme LDP, CR-LDP utilise des sessions TCP entre les LSR, au cours desquelles il envoie les messages de distribution des étiquettes. Ceci permet en particulier à CR-LDP d'assurer une distribution fiable des messages de contrôle. La figure suivante montre l'enchainement des étapes de rétablissement d'un LSP entre LRS A et LSR B en utilisant CR-LDP.

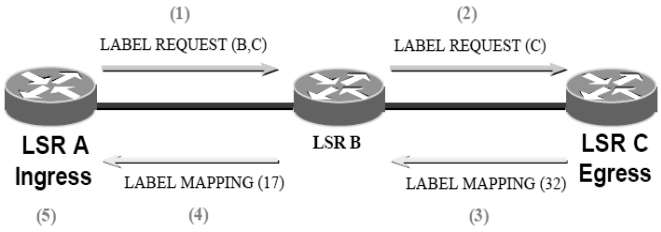


Figure 1.16: la signalisation par CR-LDP.

1.4.1.2 Le protocole RSVP TE

Le protocole RSVP avait été conçu initialement pour l'échange des messages nécessaires à la réservation de ressources pour les flux IP à travers le réseau. Une version étendue de ce protocole, RSVP-TE, permet actuellement au RSVP à être utilisé pour distribuer des étiquettes MPLS ; en d'autre termes pour permettre d'établir des tunnels LSP. RSVP TE a trois fonctions de base : l'établissement et la maintenance des chemins (Path setup and maintenance), la suppression des chemins (Path teardown) et la signalisation des erreurs (Error signalling).

La figure suivante montre l'enchainement des étapes de rétablissement d'un LSP entre les LSR E et H.

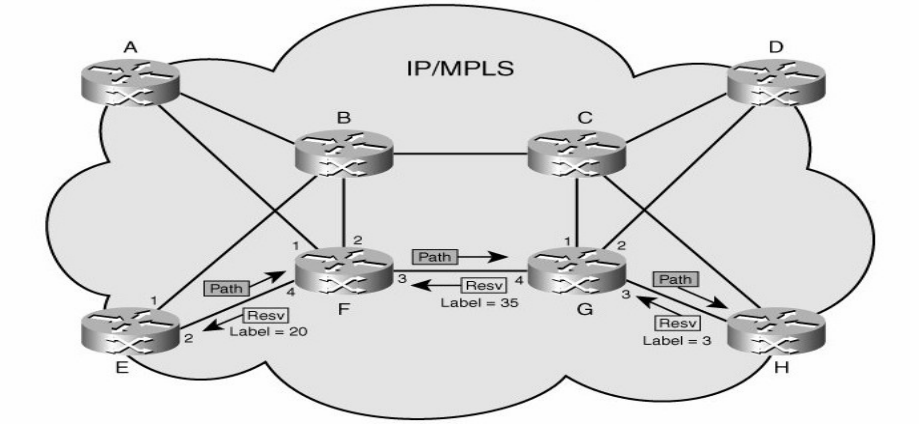


Figure 1.17: la signalisation par RSVP TE.

1.4.2 MPLS et DiffServ

MPLS prend en charge DiffServ mais avec de petits ajustements. MPLS n'introduit aucune modification à la gestion de trafic et les concepts de PHB définis dans DiffServ. Les LSR utilisent les mêmes mécanismes de gestion du trafic (metering, marking, shaping, policing, queuing ...) pour implémenter les différents PHB.

Le code DSCP de DiffServ contient 6 bits: les premiers 3 bits désignent la classe d'ordonnancement (PSC : PHB Scheduling Class), à savoir: EF, AF1, AF2, AF3, AF4 et BE, et les 3 derniers désignent la priorité de suppression (Drop Precedence) au sein de la classe comme AF11, AF12 et AF13 dans AF1.

Le standard [RFC3270] définit deux manières d'intégrer DiffServ avec MPLS:

1.4.2.1 E-LSP (EXP-Inferred PSC LSPs)

Le champ EXP est utilisé seul pour désigner le PHB. C'est-à-dire seulement 8 PHBs sont possibles. Le mappage entre DSCP et EXP est défini soit de manière fixe dans tout le domaine, soit de manière dynamique à l'établissement de chaque LSP par le protocole LDP ou RSVP. La figure 1.18 montre l'affectation d'EXP suivant la table de mappage DSCP-EXP.

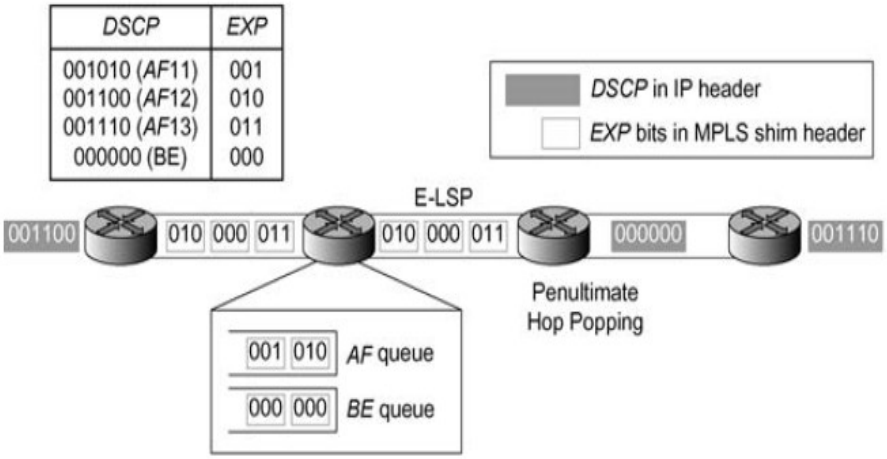


Figure 1.18: le mappage DSCP-EXP.

1.4.2.2 L-LSP (Label-Only-Inferred-PSC LSPs)

Avec ce mode de LSP, le DSCP est divisé en deux parties de tel sort que les bits représentant les informations d'ordonnancement PSC sont mappé dans le label et les bits représentant les informations sur la priorité de suppression (Drop Precedence) sont copiés dans le champ EXP. La table de correspondance PSC-Label est définie lors de l'établissement de l'LSP. Une modification a été portée sur les protocoles LDP et RSVP afin de supporter cette signalisation [Braun]. La figure 1.19 montre l'affectation des labels suivant le PSC, et l'EXP suivant le DP.

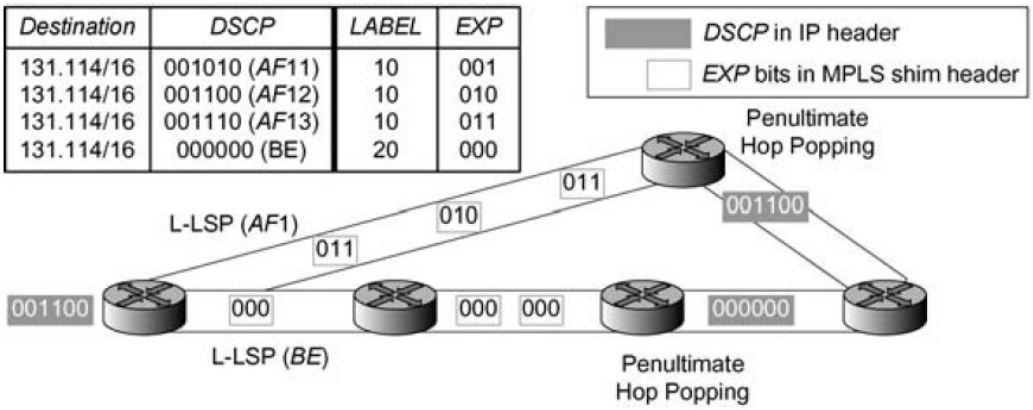


Figure 1.19: affectation de Label-EXP suivant PSC-DP.

1.4.3 DiffServ-Aware Traffic Engineering (DS-TE)

L'objectif de DS-TE est de permettre la réservation de la bande passante par classe de trafics. DS-TE consiste à diviser la bande passante réservable en *class type :* CT, CT0, CT1,... CT7 (8 CT sont définis). Chaque CT a un pourcentage de la bande passante et un niveau de priorité. Les flux affectés à un CT utilisent la bande passante associée à celui-ci. Néanmoins, et en cas de saturation de bande passante, Ils peuvent ou ne peuvent pas utiliser la bande passante des autres CT selon le mode de contrainte de bande passante BC (bandwidth constraint). Deux modèles de BC sont définis :

- *Maximum allocation model* (MAM) : une quantité de la bande passante est dédiée à chaque CT, et il ne peut pas utiliser la bande passante des autres CT même s'elle est non utilisée.

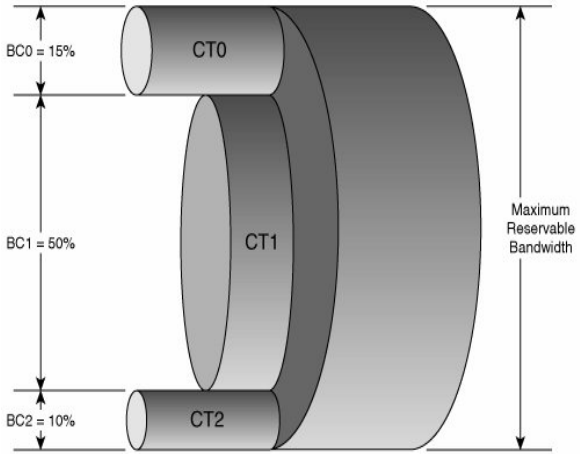


Figure 1.20: le modèle d'allocation maximum (MAM).

- *Russian Dolls Model* (RDM) : Chaque classe a sa propre bande passante, mais les classes de priorité élevée peuvent utiliser la bande non utilisée des classes moins prioritaires.

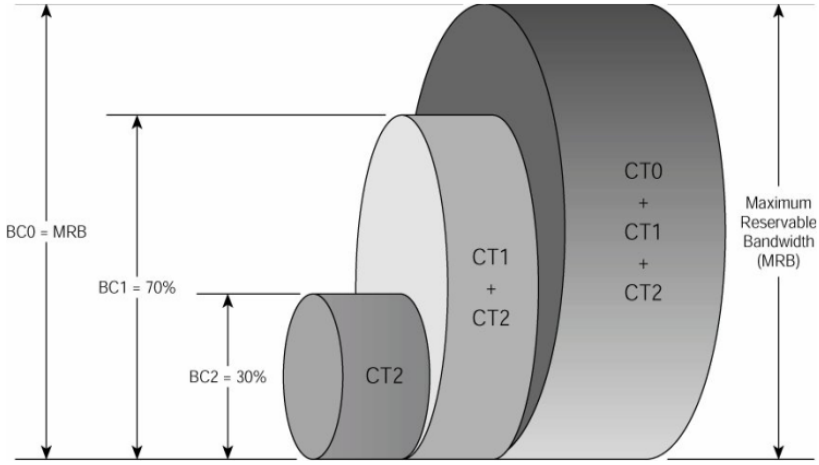


Figure 1.21: le modèle d'allocation Russian Dolls (RDM).

Pour calculer un route LSP, CSPF doit connaitre la bande passante disponible de chaque CT. Une fois l'LSP est calculé et établi, le DiffServ est utilisé pour lui donner le traitement (PHB) adéquat au niveau des LSR [Zhang].

1.5 Policy-Based Network Management (PBNM) et TE

IETF a défini une architecture capable de d'intégrer le TE dans un système de gestion de réseau par règles (PBNM). La figure 1.22 illustre les éléments de cette architecture.

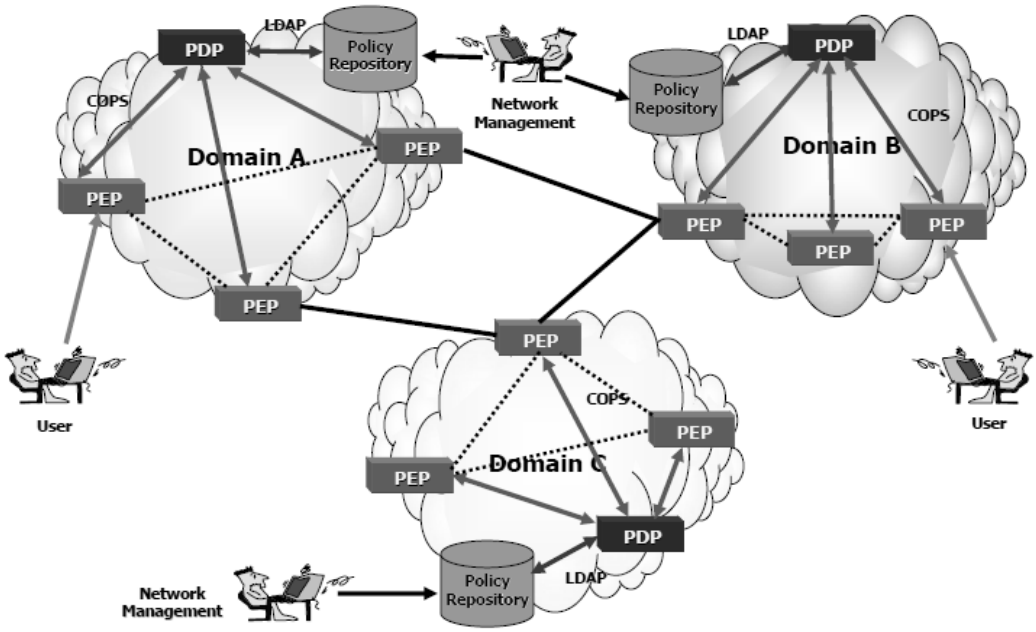


Figure 1.22: architecture de PBNM et TE.

Les PEP (Policy Enforcement Points), comme routeurs, commutateurs, points d'accès…, sont les éléments où s'appliquent les mécanismes de TE. Les actions à appliquer sont stockées dans les Policy Repositories sous forme de règle: SI (conditions à satisfaire) ALORS (action à exécuter). Le PDP (Policy Decision Point) est un élément responsable de sélectionner la règle adéquate à la situation et de l'appliquer sur les PEP appropriés. La règle (l'action à appliquer) est transférer à l'PEP par le protocole COPS (Common Open Policy Service). Outre la QoS, le PBNM est conçu pour gérer des autres aspects de réseau tel que la sécurité, le Billing, ….

L'architecture PBNM vise à standardiser, à la fois, l'interface entre les éléments de réseau (routeurs, commutateurs, …) et le système de mesure afin de déterminer l'état de réseau, et l'interface entre le système de décision (système TE dans ce cas) et les éléments de réseau afin de faciliter la configuration de mécanismes implémentés dans ces éléments.

Dans cette architecture, l'état actuel de réseau est mesuré périodiquement et comparé à l'état désiré. Ces informations ou la différence entre ces états sont utilisées par un système TE de décision pour prendre une décision sur les actions à entreprendre (figure 1.23).

Il est à noter que la période de contrôle et la décision à prendre sont en relation étroite. Pour une longue période, le contrôle (mesure) se focalise sur le dimensionnement et la planification de réseau. Par exemple, l'observation (ou l'estimation) d'une charge supplémentaire significative pour une longue période entraîne le redimensionnement de réseau qui a un coût élevé. Pour une période moyenne, la décision intervient au niveau de routage et de partage de charge. Par exemple, l'ajustement de métrique d'un protocole IGP ou de TE dans un réseau MPLS TE. A courte période, le contrôle est au niveau de flux d'utilisateur et la décision intervient au niveau de mécanismes de contrôle d'admission, de Chaping, etc. Par exemple, dans un réseau implémentant l'architecture de QoS DiffServ, l'observation de l'insatisfaction d'un client entraîne le changement de sa classe de service, et par conséquence, son code DSCP.

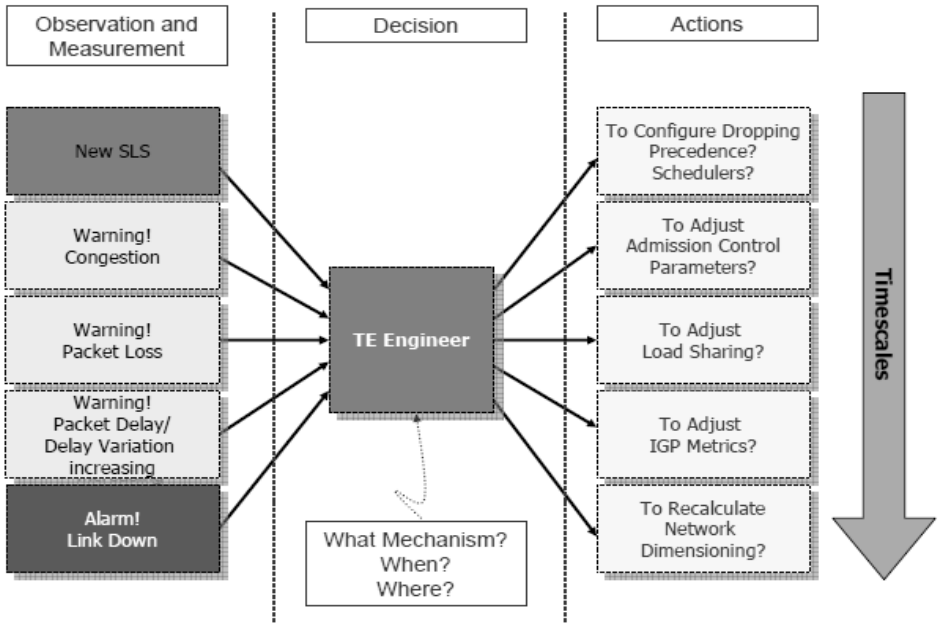


Figure 1.23: contrôle par un système TE.

1.6 Routage à QoS

Le routage à QoS est défini dans [RFC 2386] comme « un mécanisme de routage par lequel les routes des flux sont déterminées en connaissant à la fois les ressources disponibles de réseau et la QoS requise par le flux ». Il a pour objectif de:

- répondre au besoin de QoS de l'utilisateur par la recherche de la route de source à la destination qui satisfait les besoins de QoS en termes de bande passante, délai de bout en bout, etc.
- optimiser l'utilisation de ressources de réseau. Il devrait distribuer les flux de manière à maximiser le Throughput de réseau.

1.6.1 Les défis

La conception et l'implémentation de routage à QoS est plus difficile que le routage « Best-effort ». Plusieurs solutions ont été proposées, mais le point le plus important à compter est que la solution devrait être applicable avec un coût acceptable (la complexité en temps et en espace, la charge supplémentaire de trafics). Les principaux points de défis sont:

1.6.2 Métrique et calcule de route

Le problème basique de routage à QoS est premièrement comment collecter, mesurer, et diffuser les informations pertinentes sur l'état de réseau (bande passante disponible, délai, etc.), et puis comment calculer la route « faisable ou optimale » à partir de ces informations en réduisant la complexité des algorithmes en temps et en espace. Le choix de métrique est très important parce qu'elle doit représenter une propriété intéressante de réseau comme la bande passante disponible, délai, etc. Il est impossible de supporter un besoin de QoS exprimé en métriques ne pouvant pas être mappés en métrique existant de QoS.

Le calcul de route qui répond à une combinaison de critères est un problème NP-complet. Le problème NP-complet est un problème de décision dont : (i) la vérification d'une solution est rapide (en temps polynomial), et (ii) la trouvée d'une solution est difficile (en temps exponentiel).

Trois approches de calcul existent:

- pré-calcul : les routes sont calculées d'avance et mises à jour conformément à l'état de réseau;
- on-demande : la route est calculée à la réception d'une requête;
- hybride : combiné des approches précédent de façon que le calcul on-demande est déclenché seulement si les routes calculées d'avance ne satisfaisant pas la requête [Frikha].

L'approche la plus répandue est l'approche on-demande [Bruin]. Cependant, cette approche souffre de deux inconvénients : (i) elle introduit un délai supplémentaire avant le commencement de l'acheminement de paquets, et (ii) dû à l'exécution de processus de calcul à chaque demande de connexion, elle ajoute une charge supplémentaire sur les routeurs et particulièrement dans le cas où le taux d'arrivé des requêtes est élevé.

Le calcul de route est lié à la réservation de ressources. Une fois la route est sélectionnée, les ressources nécessaires sont réservées au long du chemin allant de la source à la destination. Après la réservation, chaque routeur recalcule la quantité de ressources disponible, et les autres routeurs sont avertis de ces changements. De cette manière, le calcul de route est fait sur base des informations réelles sur l'état de réseau.

1.6.3 Propagation des informations de routage

Le routage QoS nécessite l'échange de plus d'informations, en taille et en fréquence, que le routage « Best-effort ». Premièrement il compte, outre les métriques de routage BE, plusieurs autres métriques qui représentent les paramètres de QoS. De plus, ses nombreuses métriques sont susceptibles de fréquents changements dus à la réservation et à la libération de ressources. Si ces informations sont diffusées à chaque changement, ils vont produire une charge supplémentaire sur les liens et les routeurs de réseau. Mais si on diffère la diffusion, le calcul de routes se serait effectué sur des informations imprécises.

Une solution consiste à définir un seuil pour distinguer les changements significatifs, et ainsi les informations ne soient pas diffusées tant qu'elles n'atteignent pas ce seuil. Cette solution est un compromis entre la pertinence et l'efficacité.

Les protocoles qui utilisent les messages de sondage (probe messages) pour collecter des informations sur l'état de liens posent une charge additionnelle.

Vu le nombre colossale de nœuds et de liens dans Internet et sa conséquence sur la complexité de calcul de route et sur la quantité des informations à échanger, le routage à QoS recourt à l'agrégation hiérarchique de nœuds ou de domaines pour réduire la quantité des informations à sauvegarder et à maintenir.

1.6.4 L'imprécision des informations de routage

Dans le réseau étendu, maintenir des informations instantanément précises sur l'état de réseau qui se change dynamiquement est presque impossible. En fait, la plupart des informations de routage dans Internet ne reflètent pas l'état instantané de réseau [Labovitz]. L'imprécision provient de:

- la fréquence de changement de paramètres et de métriques associés aux nœuds et aux liens est plus élevée que la périodicité de l'envoi des messages de mise à jour des informations de routage. La topologie de réseau se change de manière moins fréquente (figure 1.24);
- la propagation de ces informations pend un délai non négligeable jusqu'à la convergence de réseau vers un état unifié [Yuan];
- l'utilisation de l'estimation pour déduire l'état de réseau;
- l'impact de mécanisme de mesure utilisé [Bruin].

Une solution proposée consiste à utiliser des approches de l'intelligence artificielle pour optimiser les algorithmes de routage prenant en considération, à la fois, l'état actuel de réseau et son évolution [Mellouk].

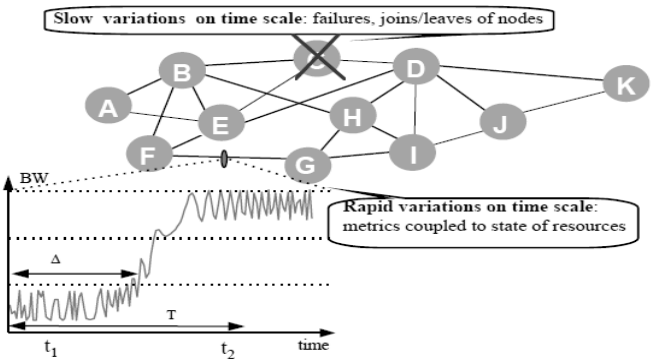


Figure 1.24: fréquences de changement différentes.

1.7 Algorithmes de routage à QoS

Le routage, en générale, implique deux éléments: protocole de routage et algorithme de routage. Le protocole de routage gère la dynamicité de processus de routage: capture l'état de réseau et ses ressources disponibles, et puis diffuse ces informations à travers tout le réseau. L'algorithme de routage utilise ces informations pour calculer les routes répondant aux critères requises. Les métriques de QoS sont classifiées en trois types: concave comme la bande passante, additive comme le délai et multiplicative comme la probabilité de perte. Les métriques multiplicatives peuvent être transformées en additives métriques par l'utilisation de fonctions logarithmiques [Mieghem].

1.7.1 Le problème Multi-Constrained (Optimal) Path, MC(O)P

Le problème général de calcul de chemin répondant à des contraintes additives est appelé : *multi-constrained path* (MCP). Le problème MCP est formulé comme suite :

Soit un réseau représenté par le graphe $G(N, E)$ où N est l'ensemble de nœuds et E est l'ensemble de liens. Soit m le nombre de métriques. Chaque lien est caractérisé un vecteur de m dimensions constitué de m non négatives métriques de QoS $(w_i(u, v), i = 1..m, (u, v) \in E)$.

Les métriques concaves (min, max) sont traitées facilement par l'élimination des liens ne satisfaisant pas la contrainte min (max) de QoS demandée. Ce traitement est appelé *topology filtering*.

1.7.1.1 Définition du problème MCP

Soit un réseau $G(N, E)$ où à chaque lien $(u, v) \in E$ est associé un vecteur de m additives métriques de QoS $w_i(u, v) \geq 0, i = 1..m$. Et soit m contraintes $L_i, i = 1..m$. Le problème MCP consiste à trouver un chemin (path) P du nœud source s au nœud distinataire d tel que :

$$w_i(P) = \sum_{(u,v)\in P} w_i(u,v) \leq L_i, i = 1..m.$$

Un chemin satisfaisant cette condition est appelé chemin faisable (*feasible path*). Aucun ou plusieurs chemin faisables peuvent exister. Il serait mieux de trouver le chemin le plus court $l(P)$ de cet ensemble. C'est le problème *multi-constrained optimal path (MCOP)*.

1.7.1.2 Définition du problème MCOP

Soit un réseau $G(N,E)$ où à chaque lien $(u,v) \in E$ est associé un vecteur de m additives métriques de QoS $w_i(u,v) \geq 0, i = 1..m$. Et soit m contraintes $L_i, i = 1..m$. Le problème MCOP consiste à trouver un chemin P du nœud source s au nœud distinataire d tel que :

(i) $w_i(P) = \sum_{(u,v)\in P} w_i(u,v) \leq L_i, i = 1..m$;
(ii) $l(P) \leq l(P^*), \forall P^*$ satisfaisant (i) ; où $l(P)$ est la *fonction d'objectif* de P.

Le problème MC(O)P est NP-complet. C'est-à-dire, il n'existe pas une fonction polynomiale de N et E qui borne le temps nécessaire pour résoudre le problème *exactement* (par contre aux solutions heuristiques).

Vu la complexité du problème MCP, on distingue deux cas particuliers où on a seulement deux métriques : Bandwidth Restricted Path (BRP), Restricted Shortest Path (RSP).

BRP désigne l'ensemble de problèmes de deux contraintes où la bande passante est l'une des contraintes à satisfaire. Il consiste à chercher les chemins répondant au premier critère (l'ensemble de chemins faisables) et puis d'en chercher le chemin optimal conformément au deuxième critère. Les algorithmes qui résolvent le problème BRP sont : Widest-Shortest Path (WSP) et Shortest-Widest Path (SWP).

L'objectif de WSP est de trouver le chemin le plus large (en bande passante) de l'ensemble des plus courts chemins(en nombre de sauts). Alors que SWP cherche à trouver le plus court chemin de l'ensemble des plus larges chemins.

RSP est une autre forme restreinte de MCP où seulement deux métriques additives sont considérées. Plusieurs algorithmes ont été proposés pour résoudre ce problème. On va voir quelques-uns.

1.7.2 Algorithmes pour RSP

1.7.2.1 Algorithme exacte

La solution exacte est trouvée par l'examination systématique de tous les chemins de la source s au destinataire d. Cette méthode n'est pas pratique car le nombre de chemin se grandit exponentiellement avec la taille de réseau (nombre de nœuds et de liens). Constrained Bellman-Ford (CBF) est un algorithme alternatif de la solution exacte. CBF maintient la liste de chemins de s vers tous les autres nœuds en ordre croissant pour le coût et décroissant pour le délai. Il choisit le nœud dont la liste contient un chemin satisfaisant le délai avec minimum de coût. Il parcourt les nœuds adjacents en largeur. Le processus contenue tant que le délai est satisfait et il y a un chemin à parcourir. CBF résout le problème RSP exactement, mais au prix de complexité en temps exponentielle au pire des cas [Kuipers].

1.7.2.2 Combinaison linéaire (relaxation Lagrangienne)

Cette méthode réduit la complexité du problème RSP en combinant le délai et le coût en une seule métrique, et le rendant ainsi un problème de recherche de chemin optimal par rapport à la nouvelle métrique. La nouvelle métrique est la combinaison linéaire des métriques sous la forme:

$$w(u,v) = \alpha d(u,v) + \beta c(u,v)$$

où α et β sont les multiplicateurs de Lagrange. La question importante ici est comment déterminer les valeurs de α et β afin que le meilleur chemin relatif à la métrique combinée soit un chemin faisable relatif aux métriques natives.

L'algorithme *LAgrange Relaxation based Aggregated Cost* (LARAC) cherche à trouver le chemin minimal qui ne dépasse pas la contrainte de délai. Soient $c(u,v)$ et $d(u,v)$ le délai et le coût du lien (u,v); Δ_d la contrainte de délai à respecter.

Le problème de minimisation est:

$$\min\{c(P) = \sum_{(u,v)\in P} c(u,v)\,, P \in P(s,t) et\ d(P) \leq \Delta_d\}$$

En utilisant la relaxation lagrangienne, LARAC cherche à minimiser la fonction: $\{c(P) + \lambda(d(P) - \Delta_d)\}$

La méthode applique le processus de la relaxation lagrangienne qui consiste à chercher itérativement le meilleur chemin conformément à la métrique combinée et à réajuster, à chaque fois, les multiplicateurs de Lagrange [Juttner]. Les métriques considérées sont le délai et le coût. LARAC s'exécute en temps polynomial mais ne donne pas la meilleure solution [Curado].

Dans tous les cas, la minimisation de métrique combinée peut ne mener à aucune solution faisable si les métriques sont non-corrélées. Ce qui mène à la combinaison non linéaire [Curado].

1.7.2.3 Combinaison non linéaire

Le problème posé par la combinaison non-linéaire est qu'un sous chemin d'un chemin optimal n'est pas nécessairement un chemin optimal [Guo]. La figure 1.25 montre bien cette problème où la fonction de poids est : $l(P) = max(c(P)/\Delta_c\,, d(P)/\Delta_d)$. Pour le nœud intermédiaire u, le chemin P_2 est le meilleur $(l(P_1) = \max(10/12\,, 1/12) = 10/12\ ;\ l(P_2) = \max(5/12\,, 5/12) = 5/12)$, alors que le chemin entier qui passe à travers P_1 et son poids est égale à 11/12, est meilleur que celle qui passe à travers P_2 et son poids est égale à 15/12.

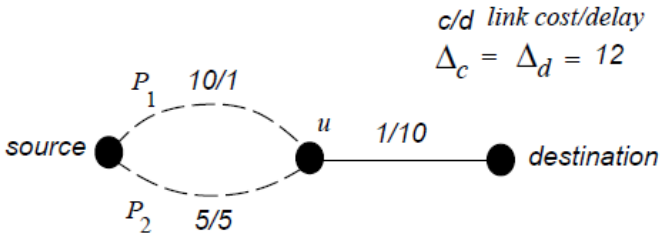


Figure 1.25: sous chemin non optimal forme un chemin optimal.

Ceci signifie que il faut stocker dans les nœuds intermédiaires, non seulement le plus court sous chemin, mais aussi tous les autres sous chemin quelque soit leur nombre.

L'algorithme *K-shortest path* est similaire à l'algorithme de Dijkstra, mais au lieu de stocker dans les nœuds intermédiaires seulement le nœud précédent et son coût optimal depuis la source, il stocke les K premiers plus court chemins et ses coûts.

L'algorithme Delay-Cost Constrained Routing (DCCR) utilise une fonction de combinaison non-linéaire. DCCR, et son extension Search Space Reduction DCCR (SSR-DCCR), cherche à trouver le chemin de moins coût mais limité par un délai Δ_d. La solution consiste à trouver, premièrement, le chemin de moins délai, et soit Δ_c son coût. L'algorithme élimine les chemins ayant un coût plus de Δ_c. En suite, Il applique l'algorithme *K-shortest path* qui retourne K chemins menant de source s au destinataire d avec un minimum de coût. Le coût d'un chemin P est calculé par la formule suivante [Guo]:

$$w(P) = \begin{cases} \dfrac{d(P)}{1 - c(P)/\Delta_c}, si\ d(P) \leq \Delta_d ou\ c(P) \leq \Delta_c \\ \infty, Autrement \end{cases}$$

Où $c(P) = \sum_{(u,v)\in P} c(u,v)$ et $d(P) = \sum_{(u,v)\epsilon P} d(u,v)$.

L'algorithme Tunable Accuracy Multiple Constraints Routing Algorithm (TAMCRA) définit une autre fonction de combinaison donnée sous la forme : $l(P) = max(c(P)/\Delta_c, d(P)/\Delta_d)$, et utilise l'algorithme *K-shortest path* pour trouver le plus courts chemin [Neve].

1.7.3 Algorithmes pour MCP

1.7.3.1 Algorithme de Jaffe

L'approximation de Jaffe est basée sur la relaxation lagrangienne pour minimiser la combinaison linéaire des métriques sous la forme de : $w(u,v) = \sum_{i=1}^{m} d_i w_i(u,v)$

où d_i sont des multiplicateurs positifs, avec la contrainte : $w_i(P) \leq L_i$. Pourtant, et comme mentionné auparavant, la combinaison linéaire peut donner un chemin qui viole les contraintes. Pour cela, Jaffe a proposé une fonction non linéaire de la forme $f(P) = \sum_{i=1}^{m} \max(w_i(P), L_i)$ dont la minimisation assure de trouvé un chemin faisable s'il existe. La complexité en temps de l'algorithme de Jaffe est évaluée à $O(NlogN + mE)$ [Kuipers].

1.7.3.2 Algorithme de Chen

L'algorithme simplifie le problème MCP par la transformation des $m-1$ métriques de valeurs réelles à des métriques entier comme suivant : $w_i^*(u,v) = \left\lceil \frac{w_i(u,v)x_i}{L_i} \right\rceil pour\ i = 2..m;$

Où x_i sont des entiers prédéfinis. L'algorithme adopte alors une approche de la programmation dynamique pour trouver un chemin qui minimise la première métrique (réel) tout en assurant que les autres métriques (entier) satisfaites les contraintes.

1.7.3.3 H_MCOP

Cet algorithme cherche un chemin satisfaisant les contraintes par l'utilisation de la fonction non-linéaire définie par l'algorithme TAMCRA (vu auparavant) et le concept de *Look-ahead*. Il cherche de minimiser, simultanément, la fonction d'objectif et la valeur de la métrique de coût.

L'idée de l'algorithme Look-ahead est de limiter l'ensemble de chemin à parcourir par l'utilisation des informations sur le sous chemin restant pour arriver à la destination. Il calcule pour chaque métrique $1 \leq i \leq m$ l'arbre de plus court chemin à partir du nœud destinataire t vers tous les nœuds $n \in N$. Il

stocke dans chaque nœud un vecteur de m éléments, où chaque élément $1 \leq i \leq m$ contient la valeur du plus court chemin de ce nœud vers le destinataire par rapport à la métrique i. On désigne par la notation $P^*_{n\to t;i}$ le plus court chemin du nœud n au destinataire t par rapport à la métrique i. $P^*_{n\to t;i}$ peut être différent de $P^*_{n\to t;j}$ pour $i \neq j$. Et soit $b(n)$ le vecteur qui contient les valeurs minimales des métriques, $b_i(n) = w_i(P^*_{n\to t;i})$.

Pour chaque nœud intermédiaire n, et pour chaque sous chemin partant de la source s vers n, la relation suivante devrait être satisfaite: $w_i(P_{s\to n}) + b_i(n) \leq L_i, i = 1..m$; où L_i est la contrainte de QoS à satisfaire sur la métrique i. En d'autre terme, si la somme de valeur de métrique i de sous chemin de s à n et la valeur minimale de i de n à t dépasse la valeur de contrainte L_i alors le sous chemin $P_{s\to n}$ viole la contrainte de QoS et ne peut pas être une solution candidate. Ainsi, l'algorithme Look-ahead réduit le nombre de chemin à parcourir dans l'espace de recherche en quête de chemin faisable. La complexité de H_MCOP est évaluée à $O(NlogN + mE)$ [Kuipers].

1.7.3.4 TAMCRA et SAMCRA

L'algorithme TAMCRA et son successeur SAMCRA se constituent de trois concepts fondamentaux:

- Une fonction de poids de chemin non-linéaire: $l(P) = max_{j=1..m}\left(\frac{w_j(P)}{L_j}\right)$;
- l'approche de l'algorithme *K-shortest path*;
- Le principe de chemin non-dominé pour réduire l'espace de recherche.

Lors de la phase de l'exécution de l'algorithme *K-shortest path*, TAMCRA ne stocke pas dans les chemins intermédiaires tous les k sous chemins venant de la source mais il fait une distinction selon le principe de non-dominance. Un chemin Q est dominé par le chemin P si $w_i(P) \leq w_i(Q)$ pour tout $i = 1..m$, avec l'inégalité pour au moins un seul i. TAMCRA stocke seulement les sous chemins non-dominé et réduit, ainsi, l'espace de recherche sans compromettre la qualité de résultat. SAMCRA ajoute le concept de look-ahead pour réduire davantage l'espace de recherche. La complexité de SAMCRA est évaluée à $O(kNlog(kN) + K^2mE)$ [Kuiper].

1.7.4 Routage inductif

L'approche inductive est basée sur l'intelligence artificielle et des techniques inspirées de la biologie comme l'apprentissage par renforcement et les algorithmes génétiques pour contrôler l'état de réseau en temps réel afin d'offrir aux utilisateurs la QoS requise [Mellouk]. Parmi ces approches on cite:

1.7.4.1 Cognitiv Packet Network (CPN)

Les CPN [Gelenbe] sont basés sur les réseaux de neurones aléatoires RNN (Random Neural Network). Dans ce type de réseau l'intelligence se situe au niveau de paquet au lieu de routeurs ou de hôtes. CPN supportent trois types de paquets: paquet intelligent, paquet d'acquittement et paquet de données. Les paquets intelligents ont la capacité de s'auto-orienter. Par l'apprentissage, ils peuvent éviter la congestion, la perte et d'être retardés. Ils s'apprennent de leurs propres observations sur le réseau de même que des expériences des autres paquets. Un paquet d'acquittement est généré par le destinataire à l'arrivée d'un paquet intelligent. Il retourne à la source du paquet intelligent en suivant le chemin inverse et met à jour les informations stockées par les paquets intelligents dans les routeurs. Les paquets de données empruntent le chemin sélectionné par les paquets intelligents.

L'inconvénient majeur des algorithmes basés sur les CPN est le temps très important nécessaire à la convergence de réseau lorsque le réseau est très chargé [Mellouk].

1.7.4.2 Optimisation par colonies de fourmis

Cet algorithme est inspiré du comportement de la colonie de fourmis pour trouver le plus court chemin menant à la source de nourriture. La table de routage contient plusieurs sorties pour chaque destination au lieu d'une seule. A chaque sortie est associée la possibilité d'emprunter le plus court chemin en sélectionnant cette sortie. Le nœud source envoie des paquets suivant les possibilités qu'il a dans sa table de routage. Ces paquets parcourent les chemins de réseau jusqu'à l'arrivée à la destination. Ils retournent ensuite sur le même chemin. A son retour, le paquet met à jour les tables de routages des routeurs se trouvant au long du chemin en augmentant la possibilité de sorite dont il en revient, et diminuant les possibilités des autres sorties. Toutefois, cette approche génère plus de trafic que des autres approches [Mellouk].

1.7.4.3 Notion de l'apprentissage par renforcement

L'apprentissage par renforcement consiste à faire apprendre un agent autonome le comportement de s'adapter lors de son interaction avec son environnement afin d'atteindre des objectifs sans aucune intervention extérieure ou d'un professeur. L'agent reçoit des stimuli de son environnement et réagit par l'exécution l'action adéquate à son comportement. Sa réaction est alors jugée par rapport à un objectif prédéfini sous forme de note « récompense ». L'agent reçoit cette récompense et l'intègre pour adapter ses actions de futures et parvenir, ainsi, à un comportement optimal. L'action qui donne une note négative devient moins préférée qu'une action notée positivement dans les mêmes conditions.

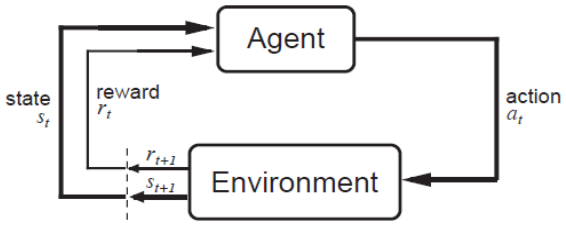


Figure 1.26: apprentissage par renforcement.

L'idée fondamentale de l'apprentissage par renforcement est d'améliorer une politique courante après chaque interaction avec l'environnement. Il s'agit d'un renforcement local qui ne nécessite donc pas une évaluation complète de la stratégie. En pratique, la plupart des algorithmes d'apprentissage par renforcement passent par l'approximation itérative d'une fonction de valeur, issue de la théorie des Processus de Décision Markovien (PDM) [Garcia].

1.7.4.4 L'algorithme Q-Routing

Cet algorithme recherche le plus court chemin en termes de temps d'acheminement des paquets jusqu'à leurs destinations. Chaque nœud x maintient une table des valeurs Q(x,y,d), appelé Q-table, où d est un élément de N, l'ensemble de tous les nœuds dans le réseau. y représente un élément de N(x), l'ensemble de tous les voisins du nœud x. La valeur Q(x,y,d) peut être interprétée comme le meilleur temps estimé par le routeur *x* pour qu'un paquet atteigne la destination *d* en passant par le routeur *y*. Ce temps n'inclut pas le temps d'attente dans la file d'attente de *x* mais inclut le temps de transmission δ, le temps d'attente dans la file d'attente de y, et le temps que le paquet prend pour atteindre *d* à partir du routeur *y* et en passant par le routeur *z* voisin de *y*.

L'idéal est que chaque routeur maintien une vision globale de l'état de réseau à tout moment. Cependant, on peut se rendre compte que la simple émission de cette information suffirait à saturer le

réseau. La solution proposée consiste à faire transiter le moindre possible d'informations dans le réseau en limitant la vision et l'échange d'information de routeur à ses voisins uniquement.

Comme le choix d'une route est basé sur les Q-valeurs et que ces dernières ne représentent qu'une estimation, la décision de routage n'est pas forcément optimale. Il est alors nécessaire de mettre à jour les Q-valeurs afin de prendre en compte l'état réel du réseau. Dès qu'un routeur x envoie un paquet *P* destiné au nœud *d* via le routeur voisin *y*, ce dernier renvoie un paquet de renforcement (signal de renforcement) au routeur x. Ce paquet contient l'estimation optimale *Q*(*y*,*z*,*d*) du temps restant pour arriver à la destination *d*. Quand le routeur x reçoit cette estimation, il recalcule la nouvelle Q-valeur *Q*(*x*, *y*,*d*) suivant une formule spécifique.

La méthode utilisée dans le Q-Routing, pour la mise à jour des Q-valeurs est connue sous le nom d'exploration avancée (forward exploration), où à chaque saut du paquet P(s,d), une Q-Valeur est mise à jour comme montre la figure 1-27 [Hoceini].

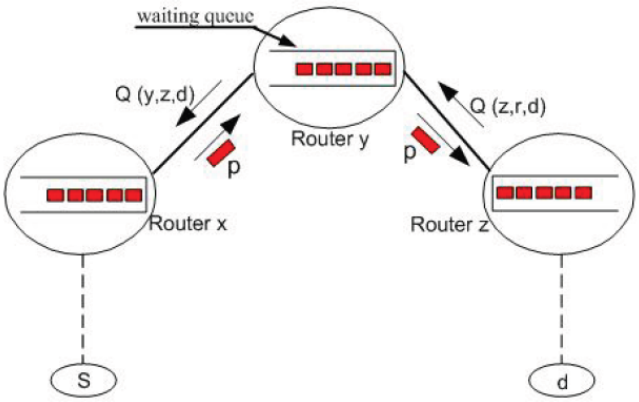


Figure 1.27:Mise à jour de Q-Valeur.

Les performances de cette politique de routage dépendent principalement des Q-Valeurs estimées qui doivent être les plus représentatives possibles de l'état courant du réseau. Par conséquence, ces valeurs doivent être mises à jour de façon continuée. Cependant, pour celles qui ne sont mises à jour que rarement, les décisions de routage sont peu fiables.

L'algorithme CQ-Routing est une amélioration de Q-Routing en attachant des valeurs de confiance (C-Valeurs) à chacune des Q-Valeurs dans le réseau. Ces C-Valeurs sont utilisés pour déterminer le pas d'apprentissage utilisé pour la mise à jour des Q-Valeurs. Une autre amélioration est l'algorithme DRQ-Routing qui ajoute une autre direction d'exploration supplémentaire "backward exploration". L'algorithme CDRQ-Routing combine les techniques des algorithmes CQ-Routing et DRQ-Routing. Ainsi, à chaque saut d'un paquet P(s,d), du routeur x au routeur y, les Q-Valeurs et C-Valeurs sont mises à jour lors des phases d'exploration forward et backward.

Ces algorithmes exhaustifs ne constituent pas forcément un choix judicieux car certains chemins sont explorés inutilement, ce qui ralentit la convergence vers un routage optimal. En effet, dans un réseau, plusieurs chemins peuvent comporter des routeurs constituant des goulots d'étranglement ou contenir des boucles. En plus, Ils sont basés sur une métrique simple qui est le temps de bout en bout [Hoceini].

1.7.4.5 K-Shortest path Q-Routing

l'algorithme K-Shortest path Q-Routing qui est basé sur la technique de routage multi-chemin combiné avec l'algorithme Q-Routing où l'espace d'exploration est réduit aux k meilleurs chemins, minimisant

ainsi le nombre de sauts. Comparé au Q-Routing, cet algorithme ne nécessite qu'un espace mémoire proportionnel au produit du nombre d'adresses de destination par le nombre K des plus courts chemins. L'algorithme de recherche des K plus courts chemins est basé sur l'algorithme de Dijkstra généralisé auquel est ajouté un mécanisme de suppression de boucles. Le chemin optimal correspond à celui dont le temps d'acheminement moyen est le plus court. Le mécanisme d'exploration qu'utilise cet algorithme pour la mise à jour des Q-Valeurs repose sur une méthode hybride associant la technique de l'exploration avancée à chaque fois qu'un paquet de donnée est échangé entre routeurs, et celle de l'exploration probabiliste pour l'exploration des k-1 chemins restants [Hoceini].

1.8 Conclusion

La QoS intra-domaine était le sujet de travail de plusieurs communautés depuis longtemps. Ils ont aboutis à définir et implémenter plusieurs techniques et mécanismes qui permettent aux réseaux d'offrir la QoS. Ces solutions couvrent à la fois la gestion de trafic (classification, conditionnement, ordonnancement,...) et l'optimisation de la gestion de ressources (l'ingénierie de trafics et routage à QoS).

Ces solutions sont limitées à l'intérieur de domaine où le réseau est sous la responsabilité d'une seule autorité (généralement opérateur) qui définit ses propres règles de routage et de QoS. Pour des raisons de sécurité et de commerce, les opérateurs ne permettent pas la propagation des informations sur la topologie et les ressources disponibles de ses domaines. Le passage à l'échelle de l'inter-domaine pose donc un autre défi qui exige des solutions complémentaires permettant de relier entre les domaines tous en assurant la QoS.

2 QoS inter-domaine

Après la « maturation » de la technologie qui permet d'offrir la QoS au sein d'un domaine, beaucoup de travaux de la communauté de recherche et de standardisation se sont orienté au-delà de domaine, c'est la QoS inter-domaine. Le but de la QoS inter-domaine est d'assurer une qualité de service de bout-en-bout en passant sur plusieurs domaines éventuellement de technologies sous-jacentes différentes. Dans ce chapitre nous abordons premièrement un peu sur les systèmes autonomes et ses types d'interconnexion (section 2.1), et nous abordons également le protocole BGP qui est conçu principalement pour l'échange des informations utiles pour le routage de paquet entre les domaines (section 2.2). La section 2.3 présente l'entité Bandwidth Broker qui gère les ressources du domaine subordonné et coopère des autres BBs pour assurer la QoS sur plusieurs domaines. SIBBS et NSIS, deux protocoles de signalisation de ressources inter-domaine, sont présentés dans la section 2.4. Dans la section 2.5 on décrit les méthodes et les procédures standardisées récemment (ou encore en version Draft) pour permettre le calcul de meilleur chemin inter-domaine. Celui-ci est très important avant de commencer la réservation de ressources.

2.1 Les systèmes autonomes

Un système autonome (AS: Autonomous System) est un groupe connecté de préfixe de routage IP sous l'instance d'un ou plusieurs opérateurs, géré par une règle de routage bien déterminée. Les AS sont interconnectés via les nœuds de bordures (BN). Les opérateurs ne permettent pas la propagation des informations sur la topologie et les ressources disponibles de ses AS pour des raisons de sécurité et de commerce. Un numéro d'AS unique est attribué par l'IANA (Internet Assigned Numbers Authority) à chaque AS. Ce numéro est utilisé pour la mise en œuvre de routage inter-domaine [RFC1930].

L'internet se constitue de l'interconnexion de plus de 35000 ASs [Huston]. Dans ce mémoire, quand nous parlons de la QoS entre les domaines, les termes domaine et AS sont interchangeables.

Les systèmes autonomes se classent en trois catégories suivant leurs types d'interconnexions et leurs opérations :

- Un AS *Multihome* est un AS qui a des connexions avec plusieurs autres AS, mais ne permet pas de transférer de trafics entre ces AS ;
- Un *AS stub* est un AS qui n'est connecté qu'à un seul AS. Son trafic est soit généré de l'intérieur, soit terminé dedans;
- Un *AS de transit* est un AS qui fournit des connexions aux différents réseaux séparés.

L'interconnexion entre les AS, le Peering, s'effectue de deux manières (figure 2.1) :

- peering : deux ou plusieurs AS s'interconnectent entre eux directement et échangent les trafics ;
- transit : un AS admet de transporter le trafic d'un AS vers des autres AS. Le transit est généralement payant.

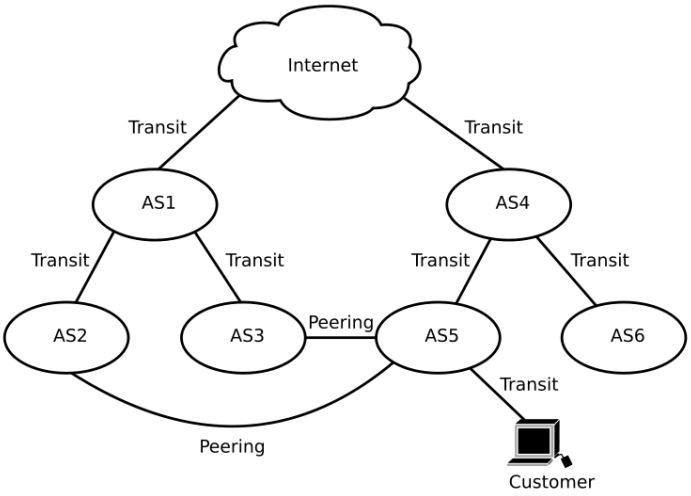


Figure 2.1: les types de l'interconnexion entre les AS.

2.2 Le Protocole BGP

BGP est un protocole de routage inter-domaine. Il supporte l'agrégation pour limiter la taille de la table de routage, et supporte également le routage sans classe CIDR (Classless Inter-Domain Routing). Les métriques ne sont pas les mêmes que celles utilisée dans les protocoles IGP car elles dépendent de la route et des politiques des opérateurs. BGP ne prend pas en compte les contraintes de QoS. La communauté de recherche explore récemment une approche distribuée pour résoudre le problème de routage inter-domaine, comme la technique BRPC (Backward Recursive PCE-Based Computation) qui utilise des entités de calcule spécifique appelées PCE (Path Computation Elements) [Frikha].

Les voisins BGP sont des routeurs avec lesquels une session BGP est établie. Ils utilisent le protocole TCP sur le numéro de port 179. BGP ne nécessite pas de mise à jour périodique des tables de routage ; des messages sont envoyés lorsque la table de routage change. En revanche, un routeur BGP doit retenir la totalité des tables de routage courantes de tous ses pairs durant le temps de la connexion. BGP est constitué de deux parties : quand il est exécuté entre deux routeurs au sein du même système autonome, il s'agit d'IBGP (Interior BGP) ; d'autre part, EBGP (Exterior BGP) est le nom donné quand la session est établie entre système autonomes différents.

2.3 Bandwidth Broker

La notion de Bandwidth Broker (BB) a été définie dans [RFC2638], dans le cadre de DiffServ, comme étant l'entité qui a la connaissance des capacités de domaine DiffServ en termes de topologie et de ressources disponibles. Il permet d'allouer de la bande passante en fonction de la disponibilité des ressources et des politiques définies pour le réseau.

Le principal module d'un BB est celui du contrôle d'admission. Il est responsable de l'admission de nouvelles demandes et de la réservation des ressources. Le BB maintient également un ensemble de données concernant les informations de topologie, de routage, des flux et de qualité de service sur chaque nœud et lien. Comme le montre la Figure 2.2 [Htira], un Bandwidth Broker est composé de :

- un module central, responsable de traitement des requêtes rentrantes. Il contrôle tous les autres modules et réalise la gestion des sessions à QoS ;
- une interface d'accès pour les différentes requêtes des applications, proxys ou administrateurs ;
- un module qui réalise le contrôle d'admission ;

- un composant en charge de la signalisation inter-domaine ;
- une base de données contenant des informations de topologie, des contrats établis avec les clients et les domaines adjacents, des politiques prédéfinies, et une description des ressources disponibles, etc.
- un système de surveillance et de métrologie.

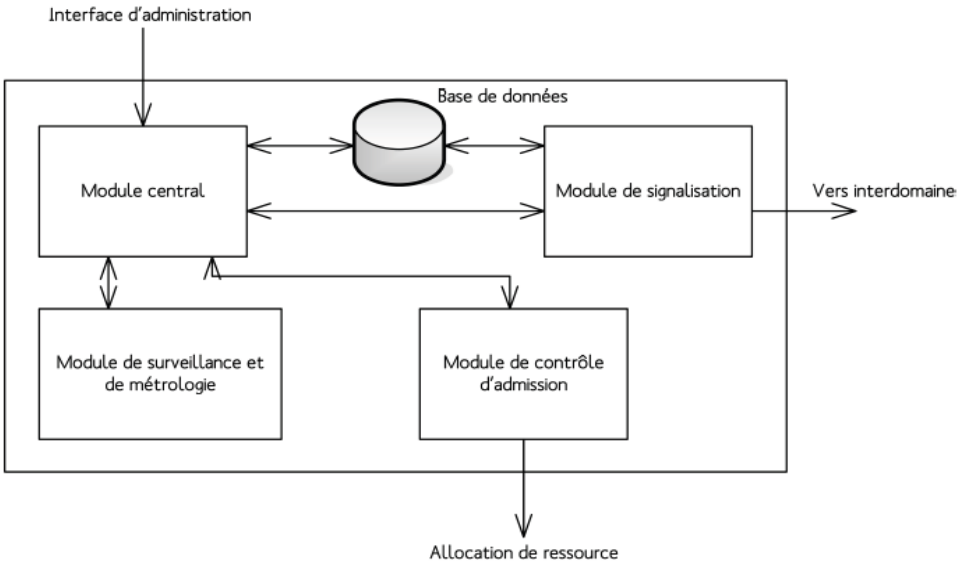


Figure 2.2: architecture de Bandwidth Broker.

Généralement un BB est déployé par domaine. Ainsi, pour que la réservation des ressources soit faite de bout en bout, les Bandwidths Brokers doivent communiquer entre eux. Pour cela des contrats de gré-à-gré (peering agreeements) sont mis en œuvre entre les domaines adjacents.

A la réception d'une nouvelle demande de connexion, un message de signalisation est envoyé par le routeur d'entrée de domaine au BB spécifiant le profil du nouveau flux ainsi que ses exigences en termes de qualité de service. Une fois la requête authentifiée, le BB prend sa décision, de refuser ou d'accepter, sur la base des politiques du domaine, les SLS établies ainsi que la disponibilité des ressources dans le réseau.

2.4 La signalisation

Dans un environnement multi domaine, la signalisation est nécessaire pour la découverte des services et de leurs performances au long du chemin de données, ainsi que pour l'évaluation de la disponibilité des ressources.

Deux axes de recherches sont suivis pour répondre aux besoins des solutions pour le provisionnement et le contrôle d'admission [Racaru] :

- une signalisation couplée au chemin de données (on-path) dont le représentant le plus abouti est RSVP ;
- une signalisation découplée du chemin de données (off-path), qui implique des équipements dédiés à la gestion de la QoS, notamment les entités de type Bandwidth Broker.

L'approche couplée au chemin de données assume une homogénéité de bout-en-bout de l'architecture Internet et le fonctionnement identique des équipements sur le chemin de données. Les

inconvénients principaux de cette approche sont le passage à l'échelle (car ils introduisent une consommation des ressources importante des routeurs) et l'hypothèse d'homogénéité des domaines.

On a déjà vu dans le chapitre précédent le protocole RSVP, qui est un exemple de la signalisation couplé au chemin de données. On va voir maintenant les protocoles SIBBS et NSIS comme exemples de protocole de signalisation inter-domaine découplé du chemin de données.

2.4.1 SIBBS

Le protocole SIBBS (Simple Inter-domain Bandwidth Broker Protocol) a été défini par le groupe de travail signalisation de QBone [QBone01]. Il a pour objectif d'être utilisé dans un environnement DiffServ guidé par un Bandwidth Broker. Dans la plate-forme QBone, chaque domaine DiffServ supporte un ou plusieurs services. Il est basé sur l'échange de deux principaux PDUs : RAR (Resource Allocation Request) et RAA (Resource Allocation Answer) (figure 2.3).

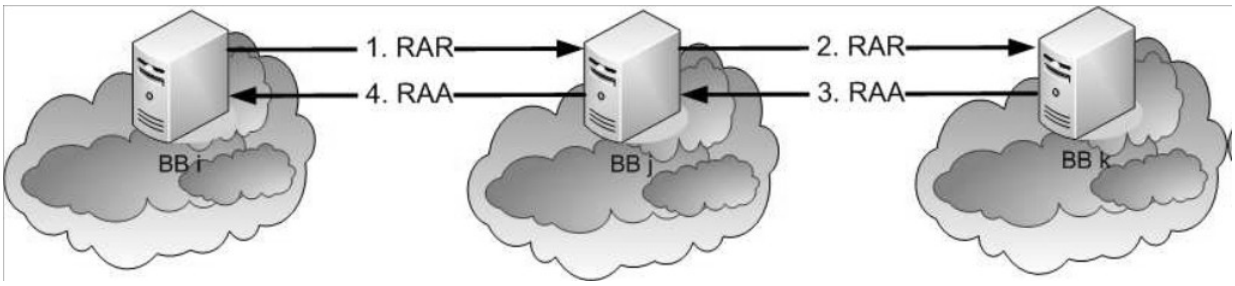


Figure 2.3: échange de messages du SIBBS

Le message RAR inclut l'identifiant de service global, des informations relatives à la requête de qualité de service (classe de service, bande passante), les adresses IP source et destination, un champ d'authentification et d'autres paramètres du service. Le message RAA contient la réponse à un PDU RAR. La communication entre les Bandwidth Brokers est supposée fiable, et utilise pour cela le protocole TCP.

Lors de la réception d'un message RAR, le Bandwidth Broker vérifie que la requête reçue respecte le SLS existant, et évalue la disponibilité des ressources à l'intérieur de son domaine pour répondre aux besoins de flux de données. Si toutes les conditions relatives aux politiques internes du domaine sont respectées il accepte cette nouvelle connexion.

Dans le cas d'acceptation, le message RAR est propagé de manière récursive vers sa destination sur tous les Bandwidth Brokers des domaines inclus dans le chemin de données. Le dernier Bandwidth Broker répond par le message RAA positif qui est acheminé jusqu'au Bandwidth Broker source, qui conclut ainsi l'admission de la requête. Le maintien des ressources est assuré par les messages de rafraichissement émis périodiquement.

2.4.2 NSIS

NSIS (Next Steps In Signaling) est un groupe de travail créé par l'IETF afin de définir un cadre générique pour la signalisation IP en tenant compte en premier lieu de la QoS mais aussi de la sécurité ou encore de la mobilité, etc…. L'architecture NSIS a été décomposée en deux niveaux [RFC4080] :

- Le niveau NTLP (NSIS Transport Layer Protocol), dédié au transport de la signalisation entre les différentes entités NSIS. Pour assurer ce rôle, un protocole nommé GIST (General Internet Signaling Protocol) a été spécifié. Son principe de fonctionnement est le suivant : le niveau GIST s'occupe de la transmission des messages de signalisation à la prochaine entité NSIS après le rétablissement

d'une association négociée en trois phases. Lorsque le message est reçu par l'entité NSIS suivante, celle-ci le fait passer au niveau NSLP pour le traitement et puis le transmet à la prochaine entité, et ainsi de suite jusqu'au récepteur final. Enfin, tout comme RSVP, GIST maintient les états de réservation au niveau des entités NSIS.

- Le niveau NSLP (NSIS Signaling Layer Protocol) spécifique à l'application de signalisation qui définit les messages et le traitement suivant son besoins.

La signalisation est effectuée saut par saut entre les entités NSIS (NE) et les dispositifs ne supportant pas NSIS acheminent simplement les messages sans les traiter.

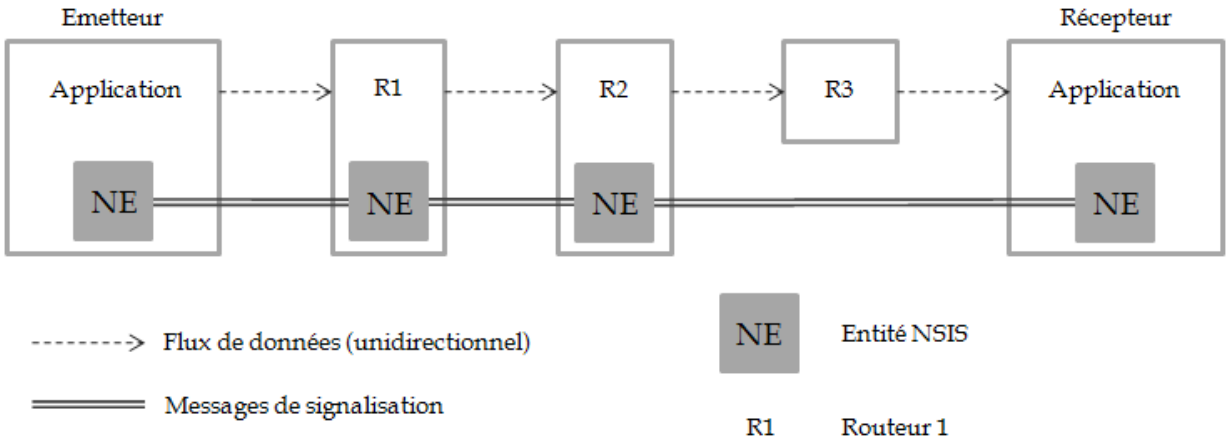


Figure 2.4: enchainement de signalisation NSIS.

2.5 Calcul de chemin multi-domaine

La figure 2.5 illustre le problème de calcul de chemin multi-domaine. Dans cette figure, le nœud A du domaine X veut communiquer avec le nœud B du domaine Z en empruntant le chemin qui satisfait certaines contraintes.

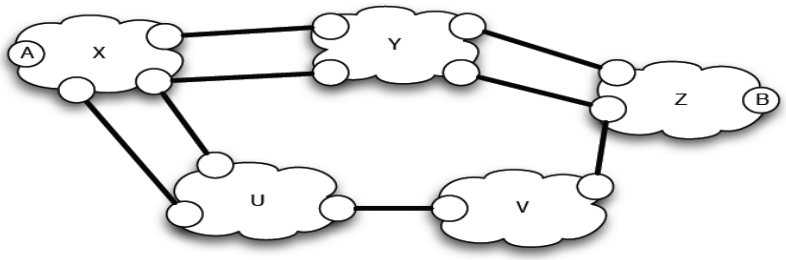


Figure 2.5: communication entre nœuds de domaines différents.

Le problème de calcul de chemin consiste à trouver un chemin de A à B passant successivement par les domaines X, Y et Z. Le calcul de chemin peut être affecté au nœud source A s'il a suffisamment d'informations sur la topologie et les ressources disponibles dans tous les domaines. Mais tant que ce n'est pas réellement le cas, deux scénarios sont possibles : calcul de chemin par domaine où le calcule est distribué sur les nœuds bordures des domaines, et calcul basé sur des entités dédiées qui se coopèrent pour effectuer le calcul. Ces entités s'appellent PCE (Path Computation Element).

2.5.1 Calcul par domaine

Le calcul par domaine (Per-domain computation [RFC5152]) s'applique quand le calcul de chemin inter-domaine ne peut pas être effectué par le nœud source due à la limite de visibilité entre les domaines. Le calcul par domaine impose sur chaque domaine intermédiaire de calculer le segment du chemin qui le concerne sans échanger aucunes informations sur le chemin avec les autres domaines. Le chemin inter-domaine complet est alors la concaténation des segments calculés par les domaines. La figure 2.6 montre les étapes de calcul par domaine où le nœud source calcule le segment menant au nœud bordure de deuxième domaine qui, à son tour, calcule le segment menant au nœud de bordure de troisième domaine, et ainsi jusqu'à l'arrivée au nœud destinataire.

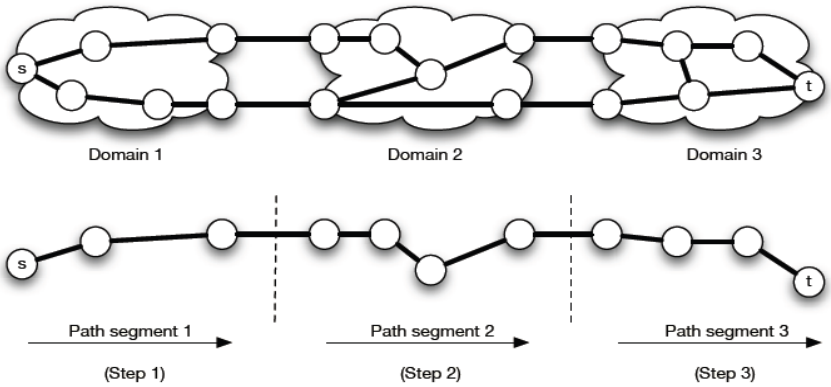


Figure 2.6 : calcul de chemin par domaine.

2.5.2 Calcul par PCE

Le PCE (Path Computation Element) est une entité capable de calculer un chemin à partir du graphe représentant le réseau en prenant en considération les contraintes imposées [RFC4655]. L'entité peut être un composant, application ou nœud de réseau [Davie]. Il démarre le processus de calcul de chemin lors de la réception de requête de la part de client PCC (Path Computation Client). Le protocole PCEP (Path Computation Element Communication Protocol) est défini pour la communication entre le PCE et le PCC, et entre deux PCE [RFC5440]. Le PCE peut calculer le chemin intra-domaine où il a suffisamment d'informations sur la topologie et les ressources disponibles dans le domaine, comme il peut coopérer avec des autres PCE pour calculer un chemin inter-domaine. BRPC (Backward Recursive PCE-based Computation) est une procédure normalisé par l'IETF qui décret la coopération entre les PCE pour calculer un chemin inter-domaine [RFC5441].

2.5.3 Backward Recursive PCE-based Computation (BRPC)

Le calcul de chemin inter-domaine est difficile lorsque la visibilité du nœud source est limitée au domaine local. BRPC implique la coopération de plusieurs PCE pour calculer un chemin inter-domaine satisfaisant les contraintes requises au long d'un enchainement de domaines du nœud source au nœud destinataire. Cette technique préserve la confidentialité des informations des domaines, un point clé lorsque les domaines sont gérés par des autorités différentes (c'est généralement le cas). BRPC se caractérise par qu'il est : (i) Backward, où le calcul de chemin commence du domaine destinataire vers le domaine source, et (ii) récursive, où la même séquence des étapes se répète sur tous les domaines intermédiaires reliant les domaines source et destinataire.

Soit une séquence de D domaines dénotés de V_1 à V_D, où V_1 représente le domaine source et V_D le domaine destinataire. BRPC définit le concept de Virtual Shortest Path Tree (VSPT). $VSPT(i)$, où $1 \leq i \leq D$, représente l'arbre multipoint à point (MP2P) et constitue des plus courts chemins menant des nœuds bordures du domaine i vers le nœud destinataire. Chaque lien de l'arbre $VSPT(i)$

représente un chemin plus court. Le PCE du V_D calcule le $VSPT(D)$ qui se compose de la liste de tous les plus courts chemins entre le nœud destinataire et les nœuds bordures de V_D. Le $VSPT(D)$ est passé ensuite au PCE du domaine $D-1$ pour calculer le $VSPT(D-1)$ et ainsi de suite jusqu'à l'arrivée au domaine source, et par conséquent, le nœud source du chemin inter-domaine [RFC5441]. La figure 2.7 montre les étapes de BRPC pour calculer un chemin inter-domaine [Bertrand].

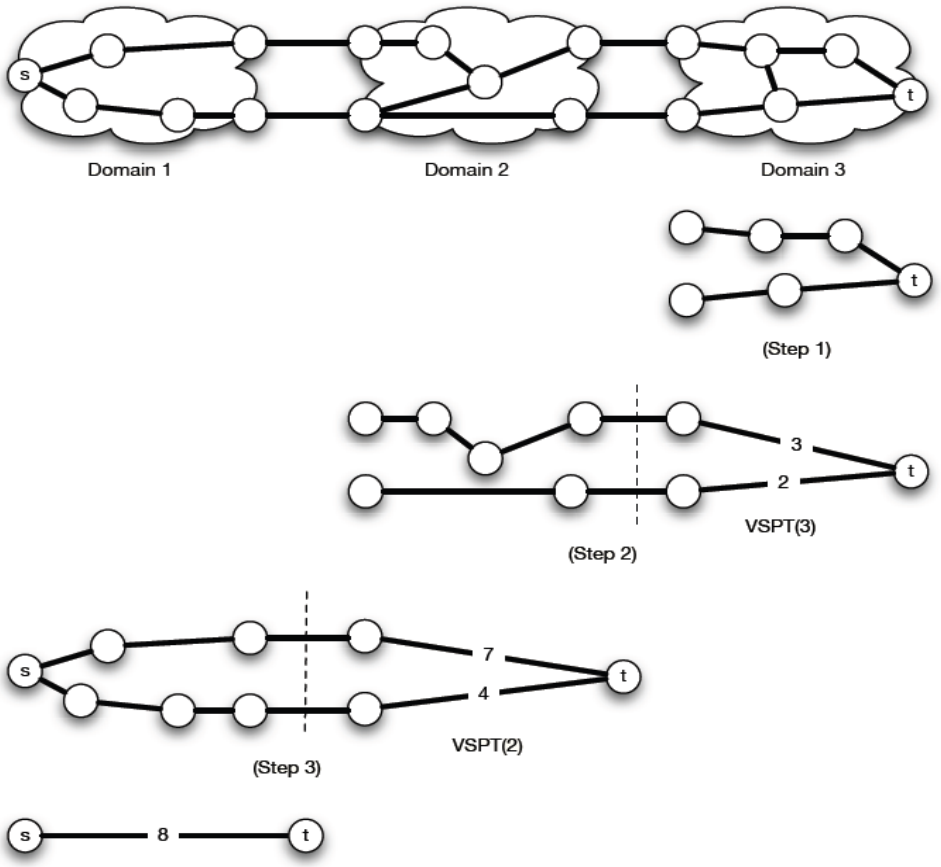


Figure 2.7: les étapes de la procédure BRPC.

2.5.4 Inter-domaine MPC (ID-MCP)

La procédure BRPC garde dans son arbre virtuel VSPT, pour chaque nœud bordure (BN), le plus court chemin entre le BN et la destination. Ceci n'est pas adéquat pour les algorithmes du problème MCP (vu précédemment). Ces algorithmes utilisent une fonction non linéaire pour calculer le coût de chemin et, par conséquence, ils ont la propriété disant que le segment d'un chemin optimal n'est pas nécessairement optimal. La figure 2.8 illustre la limite de BRPC par l'utilisation de l'algorithme SAMCRA pour calculer un chemin sous les contraintes (4,4). SAMCRA utilise la fonction de coût :

$$c(P) = Max_{i=1..K}\left(\frac{w_i(P)}{W_i}\right)$$

La ligne continue représente le plus court chemin de a à t, et il est, dans ce cas, le seul à inclure dans le VSPT selon la procédure BRPC. Son coût est donné par : $Max\left(\frac{2}{4},\frac{2}{4}\right)=\frac{1}{2}$. Le coût du chemin total de s à t en utilisant ce segment est alors $\frac{5}{4}>1$, ce qui signifie qu'il est un chemin infaisable. Alors que l'autre segment (ligne discontinue), dont le coût est égal à $\frac{3}{4}$ et qui n'est pas inclus dans le VSPT, donne un chemin total faisable avec un coût égal à $\frac{4}{4}=1$.

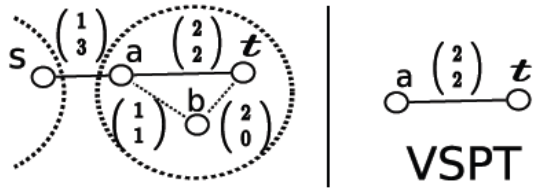


Figure 2.8: limite de VSPT.

L'algorithme ID-CMP introduit deux choses [Bertrand]:

- Extension de VSPT pour qu'il supporte plus de chemins menant d'un BN vers la destination comme montre la figure 2.9;
- Adaptation de l'algorithme TAMCRA (même que SAMCRA mais sans le concept de Look-ahead) de façon qu'il peut calculer les chemins faisables entre plusieurs sources et une destination.

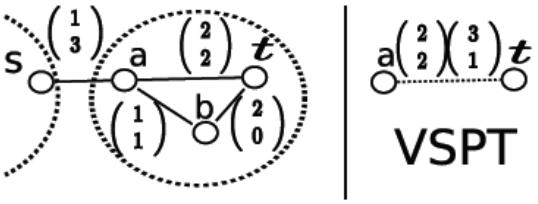


Figure 2.9 : extension de VSPT.

Dans cet algorithme, un domaine V_i concatène le $VSPT(i+1)$ et sa topologie, puis calcule pour chaque BN, tous les chemins y menant de la destination. Les chemins calculés sont inclus dans le $VSPT(i)$ et passé au domaine prédécesseur. L'opération se répète jusqu'à l'arrivée au domaine de domaine de nœud source qui concatène le VSPT reçu et sa topologie et calcule le plus court chemin inter-domaine de la source à la destination.

2.5.5 Découverte de la séquence de domaines

Dans les standards définissant le PCE et BRPC, La séquence de domaines interconnectés entre le domaine source et le domaine destinataire est supposée prédéterminée. Ils ne disent rien sur la manière de générer cette séquence malgré qu'elle ait un impact important sur l'optimalité de chemin.

Le nouveau Draft de l'IETF [PCE-H] propose à ce propos deux applications de PCE dans un scénario qui vise à trouver la meilleure séquence de domaine : Path Computation Flooding (PCF) et PCE hiérarchique.

2.5.5.1 Path Computation Flooding (PCF)

Quand la séquence de domaines est inconnue la procédure BRPC fonctionne comme suite [PCE-H] :

- le PCC (Path Computation Client) envoie une requête au PCE de domaine source (Ingress PCE) ;
- l'ingress PCE renvoie la requête directement au PCE de domaine destinataire (Egress PCE) ;
- l'egress PCE construit un egress VSPT et le diffuse sur tous les PCE des domaines adjacents ;
- chaque PCE construit, à son tour, son VSPT et le diffuse sur les PCE adjacents ;
- l'ingress PCE reçoit des VSPT de tous ses PCE adjacents, et il peut alors choisit le chemin optimal.

2.5.5.2 PCE hiérarchique

L'architecture de PCE hiérarchique définit un PCE parent qui maintient la mappe de la topologie des domaines. La mappe contient les domaines et ses interconnexions. Le PCE parent n'a aucune information sur les ressources disponibles au sein des domaines ni les états des interconnexions entre les domaines.

Quand un chemin inter-domaine est demandé, le calcul se déroule comme suite :

- l'ingress PCE envoie une requête à son PCE parent en utilisant le protocole PCEP ;
- à partir de la mappe de topologie, le PCE parent sélectionne les domaines candidats et envoie un requête aux PCE fils, responsables des domaines candidats ;
- chaque PCE fils calcule un ensemble de segments de chemins et le retourne au PCE parent ;
- le PCE parent concatène les segments et choisit le chemin inter-domaine optimal ;
- le chemin optimal est renvoyé au PCE source qui, à son tour, le renvoie au PCC qui a sollicité le chemin.

2.6 Conclusion

Les solutions présentées dans ce chapitre rendent faisable l'assurance de la QoS sur plusieurs domaines homogènes. Elles permettent de :

- la coopération des domaines dans le contexte de chercher les chemins inter-domaines faisable ;
- trouver le meilleur chemin de bout-en-bout satisfaisant les paramètres de QoS requis par un flux ;
- la réservation de ressources nécessaires pour les flux au long de chemin inter-domaine.

La limite de ces solutions provient de l'hypothèse de l'homogénéité de domaines. Ce qui n'est pas le cas réellement dans l'internet où les technologies sous-jacentes des domaines se varient grandement, aussi bien au niveau de réseaux d'accès (xDSL, Ethernet, UMTS, CDMA, Wi-Fi, WiMAX, VSAT …) qu'au niveau de réseaux de transport (MPLS, ATM, Carrier Ethernet …). Et chaque technologie est dotée de son paradigme de QoS et sa manière de l'implémenter. Qu'sera-t-il la QoS de bout-en-bout d'une communication qui traverse plusieurs domaines de technologies différentes ?

Dans le chapitre suivant nous abordons le projet EuQoS qui vient adresser le problème de l'hétérogénéité de domaine en proposant un système de qualité de service de bout-en-bout dans un environnement multi-opérateur, multiservice et multi-technologie.

3 L'architecture EuQoS

3.1 Présentation

EuQoS (End-to-end Quality of Service support over heterogeneous networks) est un projet européen du sixième Programme Cadre pour la Recherche et le Développement. La motivation principale du projet est liée à l'utilisation croissante de l'Internet comme infrastructure globale de communication et à la volonté des opérateurs d'offrir de nouveaux services à valeur ajoutée avec QoS garantie [Racaru][Mingozzi][Braun].

L'objectif principal d'EuQoS a été de concevoir, de développer, d'implémenter et de démontrer une architecture permettant de garantir la qualité de service de bout-en-bout dans un environnement le plus général possible : multi-opérateur, multiservice et multi-technologie. De plus, EuQoS a visé un éventail large d'applications nécessitant de la QoS, à savoir : voix sur IP, vidéoconférence, vidéo-streaming, télé-enseignement, télé-engineering et télémédecine. Le résultat du projet est le « Système EuQoS », déployé sur un ensemble de plates-formes de tests européennes (Testbed), hétérogènes du point de vue des technologies sous-jacentes.

Le consortium du projet EuQoS se compose de 24 partenaires : opérateurs de télécommunication (Telefonica, France Telecom, ...), corporations (Siemens, Ericssons, Juniper, Datamat ...), laboratoires de recherche et universités (LAAS-CNRS, University of Bern, Warsaw University, University of Rome ...).

Le projet EuQoS a conçu, déployé et validé une architecture globale, qui intègre un large ensemble de mécanismes divers et cohérents, totalement intégrés : autorisation, authentification, négociation de service, contrôle d'admission, signalisation, surveillance et métrologie, ingénierie de trafic et optimisation des ressources.

L'organisation des taches (Work Packages - WP) a été la suivante :

- WP 1 - Business Model and System Design: Le rôle du WP1 était de concevoir le système EuQoS en prenant en compte les technologies existantes ou en cours d'étude. D'une part, le WP1 a identifié et conçu les éléments principaux de l'architecture (les fonctions, les composants, les interfaces) et a étudié le modèle de marché (Business Model) pour créer des nouvelles relations entre les utilisateurs et les fournisseurs de réseaux et de services.
- WP 2 - Validation of the EuQoS System: L'objectif de ce WP était double: d'une part valider les capacités du système à garantir la QoS par l'intermédiaire des outils de simulations développés par les partenaires du projet ; d'autre part, proposer des outils de surveillance et de mesure pour valider l'architecture déployée sur une plateforme européenne.
- WP 3 - Implementation of the EuQoS System: L'objectif principal du WP3 a été l'implémentation de l'architecture EuQoS. Ces travaux ont été menés en plusieurs étapes qui ont conduit à 5 prototypes intégrés, déployés et démontrés lors des diverses revues.
- WP 4 - Adaptation of Applications: Les travaux de ce WP ont porté sur la conception des applications et/ou leur adaptation au système EuQoS, avec trois objectifs : (1) l'identification des besoins en QoS pour les applications ; (2) le développement des modules d'adaptation pour les applications existantes dans le but de prendre en compte le système EuQoS et (3) l'intégration dans l'architecture EuQoS de nouvelles applications multimédia déployées sur différents réseaux d'accès.
- WP 5 - EuQoS Pan European trials: il a défini le déploiement, les tests et les mesures sur une plate-forme européenne (en relation avec les WP3 et WP4). Le rôle de ce WP est d'identifier les types d'environnement réseaux à utiliser (Testbeds), de les configurer, de réaliser leur interconnexion via

une plate-forme paneuropéenne, de mener des campagnes des tests et de mesurer et analyser les résultats.
- WP 7 - Project Management : Le rôle du WP7 était la coordination interne du projet et son administration.

3.2 Notion de service réseau

La notion de service réseau, NS (Network Service), a été bien définie et implémentée dans le réseau ATM. Un service réseau est le service que le réseau offrit aux flux des applications. Les paquets utilisant un service vont être transférés conformément aux garanties bien déterminées par le service. Par exemple, les paquets bénéficiant du service BE (Best Effort) peuvent subir un délai non borné et peuvent même être perdues.

En EuQoS, le terme QoS NS désigne le service réseau qui garantit aux paquets des flux une qualité de service spécifique exprimée en forme de délai, perte etc.

La définition d'un service réseau nécessite la spécification de (figure 3.1):

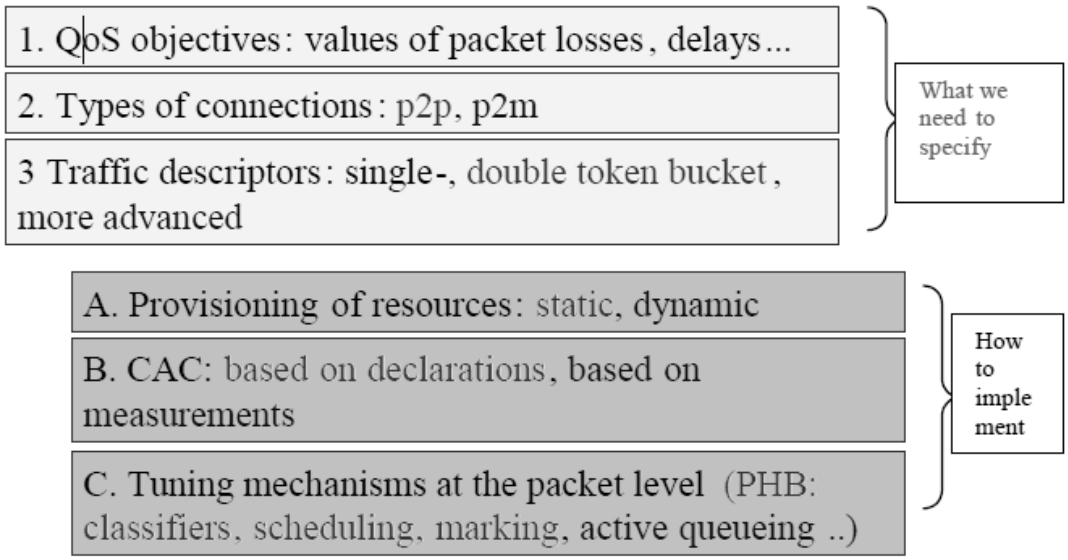


Figure 3.1: élémens de service réseau.

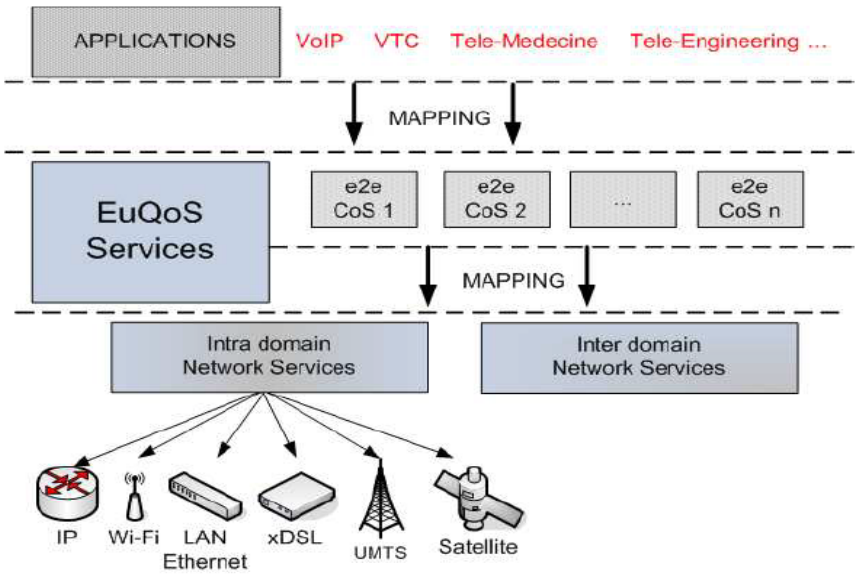
- Objectifs de QoS à garantir spécifiés en termes de paramètres ayant de valeurs concrètes. Les paramètres concernés sont :
  - Maximum de délai de transfert de paquet ; inférieur à 150 msec, par exemple.
  - Maximum de variation de délai ; ex : inférieur à 20 msec.
  - Maximum de taux de perte de paquet ; ex : inférieur à 10^{-4}.
  - Maximum de la bande passante ; ex : 100 kbps.
  - ...
- Types de connexions supportées par le service. Une connexion peut être de type point à point (p2p), point à multipoint (p2m) ou point-to-anywhere (p2e).
- Modèle de description de trafic. Parmi les modèles utilisés :
  - Single Token : utilisé pour déclarer un flux de débit constant caractérisé par le débit de crête, PBR (Peak Bit Rate), et la tolérance de PBR, PBRT (Peak Bit Rate Tolerance).
  - Double Token : utilisé pour déclarer un flux de débit variable caractérisé par deux TB (Token Backet) ; un pour déclarer le PBR et le PBRT, et l'autre pour déclarer le débit durable, SBR (Sustainable Bit Rate) et le buffer de SBR, MBS (Maximum Burst SBR).

L'implémentation de service spécifié par les trois points précédents nécessite :

- dimensionnement de service en lui attribuant de ressources (provisionnement). Précisément on lui alloue une bande passante et un espace de buffer.
- algorithme de contrôle d'admission approprié à chaque service qui contrôle l'acceptation des nouvelles connexions.
- mécanismes de QoS appliqués au niveau de paquet (classification, conditionnement, ordonnancement, ...) qui effectuent le traitement spécifique à chaque classe au niveau de routeurs.

3.3 Le modèle de QoS

Afin d'offrir la QoS sur des réseaux hétérogènes, l'approche d'EuQoS repose sur la conception d'un cadre (Framework) abstrait indépendant des technologies des réseaux. Le premier élément de ce cadre est les classes de service de bout-en-bout, e2e CoS (end-to-end CoS). Ces classes sont associées à des familles d'applications, et leurs paramètres de QoS ont une portée de bout en bout sur tous les réseaux de tous les domaines. La correspondance entre les e2e CoS et les classes de service spécifiques à la technologie sous-jacente de réseau (telles que AF et EF de DiffServ par exemple) est réalisée au sein de chaque domaine. La Figure 3.2 illustre la correspondance proposée entre les e2e CoS du système EuQoS et les CoS des déférentes technologies.



3.2: correspondance entre les CoS de déférents niveaux.

EuQoS s'appuie sur les définitions de l'IETF [4594] pour définir les e2e CoS qui sont supposées connues par les applications, et implémentées et maintenues par les fournisseurs indépendamment des types de réseau sous-jacent. Chaque e2e CoS est associé à un code DSCP unique qui la distingue dans EuQoS. La table 1 montre les e2e CoS et les codes DSCP correspondants.

e2e CoS	DSCP Name	DSCP Value
Telephony	EF	101110
Signalling	CS5	101000
RT Interactive	CS4	100000
MM Streaming	AF3x	011xx0*
High Throughput Data	AF1x	001xx0*
Standard	DF	000000

* xx ∈ {01, 10, 11}

Tableau 1: correspondance entre e2e CoS et DSCP.

Les valeurs des paramètres de QoS des e2e CoS sont définies sur tous les réseaux, et la correspondance est définie particulièrement pour les technologies étudiées dans EuQoS (xDSL, LAN, Wi-Fi, UMTS, Satellite ou réseau de cœur). Le tableau 2 résume les agrégats de trafics, la dénomination d'EuQoS, les objectifs de QoS visés ainsi que la correspondance aux applications développées dans le cadre du projet EuQoS :

Trafic Aggregate	End-to-end Class Of Service	QoS Objectives			EuQoS Applications							
		IPLR	Mean IPTD	IPDV	Nexuiz	VoIP	VTC	VoD	Medigraf			
									VTC	Collaboration	Data transfer	Chat
CTRL	Network Control	10^{-3}	100 ms	50ms								
Real Time	Telephony	10^{-3}	100/350 ms (local/long distance)	50ms		X						
	Signalling	10^{-3}	100 ms	U								
	MM Conferencing	10^{-3}	100 ms	50ms								
	RT Interactive	10^{-3}	100/350 ms (local/long distance)	50ms	X		X		X			
	Broadcast Video	10^{-3}	100ms	50ms								
Non Real Time (Assured Elastic)	MM Streaming	10^{-3}	1s non critical	U				X				
	Low Latency Data	10^{-3}	400 ms	U								
	OAM	10^{-3}	400 ms	U								
	High Throughput Data	10^{-3}	1s non critical	U							X	
Elastic	Standard	U	U	U								X
	Low Priority Data	U	U	U								

Tableau 2: les e2e CoS et ses paramètres de QoS.

3.4 Architecture générale

L'architecture EuQoS a pour fort intérêt d'offrir une solution générale, qui intègre dans une vision globale un ensemble de mécanismes : signalisation, contrôle d'admission, surveillance, ingénierie de trafic, allocation des ressources, etc. De plus, le système EuQoS, en prenant en compte les propositions et les standards existants, a développé une solution modulaire multi-niveau, où chaque composant a des fonctionnalités précises.

L'architecture du projet repose sur les principes suivants :

- La prise en compte d'un contexte général multi-domaine. Les domaines sont considérés hétérogènes de point de vue de leurs technologies sous-jacentes ;
- La gestion de domaine par une entité fonctionnelle, étendant la notion de Bandwidth Brokers, appelée Resource Manager (RM) ;
- La distinction de trois plans pour la gestion de la qualité de service : plan de service, plan de contrôle et plan de transport ;
- La définition des classes de services de bout-en-bout multi-domaine, e2e CoS, qui seront alors associées à des familles d'applications dont les valeurs des paramètres de QoS sont bien définies ;
- La définition de protocole de routage inter-domaine, EQ-BGP, prenant en compte la QoS. Il est une variante enrichie du protocole BGP standard ;
- La distinction entre les deux entités : le client EuQoS (logiciels situés dans le terminal utilisateur) et le serveur EuQoS (le cœur qui met en place les mécanismes afin de garantie de QoS tout au long du chemin des données) ;
- L'adaptation ou l'extension de protocoles standards ou en cours de développement (en particulier ceux de l'IETF) pour prendre en compte les besoins de garantie de QoS. Ainsi, dans le cadre d'EuQoS de nouveaux protocoles ont été conçus et développés, tels que EQ-SIP, EQ-NSIS et EQ-BGP.

3.5 Les composants d'EuQoS

La Figure 3.3 détaille les composants de l'architecture EuQoS, leur localisation (coté client ou serveur), ainsi que les méthodes de communication entre ces différents composants.

Le client EuQoS, localisé sur l'équipement de l'utilisateur, est constitué des modules suivants :

- l'application qui exprime des besoins en QoS ;
- le module « Application Signaling » qui permet d'échanger des informations de signalisation entre les terminaux utilisateurs ;
- le module QCM (Quality Control Module) qui prend en charge l'adaptation des paramètres au système EuQoS et l'invocation de service offert par le serveur. Tous ces composants appartient au plan de service ;
- le module « Transport Protocols » situé dans le plan de transport et dans la couche OSI transport et fournit, en plus des protocoles classiques (TCP, UDP), des mécanismes, des protocoles et des services enrichis pour le transfert de données.

Le serveur EuQoS implante les trois plans détaillés précédemment :

- le plan de service est composé de trois sous modules :
  - AQ-SSN (Application Quality Service Signaling Negotiation) qui offre l'accès au service EuQoS pour ses utilisateurs. Il fournit une interface basée sur SOAP (Simple Object Access Protocol);
  - CHAR (Charging) qui prend en charge la tarification des services ;

- SAAA qui est en charge de l'authentification, l'autorisation et la comptabilisation. Il communique les données récoltées au module de tarification (CHAR) et échange des messages avec AQ-SSN par le protocole Diameter ;

- le plan de contrôle est constitué de deux niveaux NTI et NTD :
  - Le niveau NTI comprend deux composants principaux : le Resource Manager (RM), en charge de la gestion du domaine, et le PCE (Path Computation Element), en charge de la sélection des routes. Ces deux éléments communiquent par l'intermédiaire du PCEP (PCE Protocole).
  - Le niveau NTD implante des fonctionnalités liées à la gestion spécifique de la QoS suivant la technologie sous-jacente ; il est constitué de deux modules, le Resource Allocator (RA), en charge du traitement des requêtes de QoS, et le Monitoring et Measurement System (MMS), pour assurer la surveillance et la métrologie du réseau. La communication entre les modules RM et RA s'effectue par le biais du protocole EQ-COPS.
- le plan de transport a pour but d'assurer le transfert des données sur le chemin traversant plusieurs domaines. Il présente des interfaces avec les composants du NTD (RA et MMS) et intervient dans le transfert des données du client.

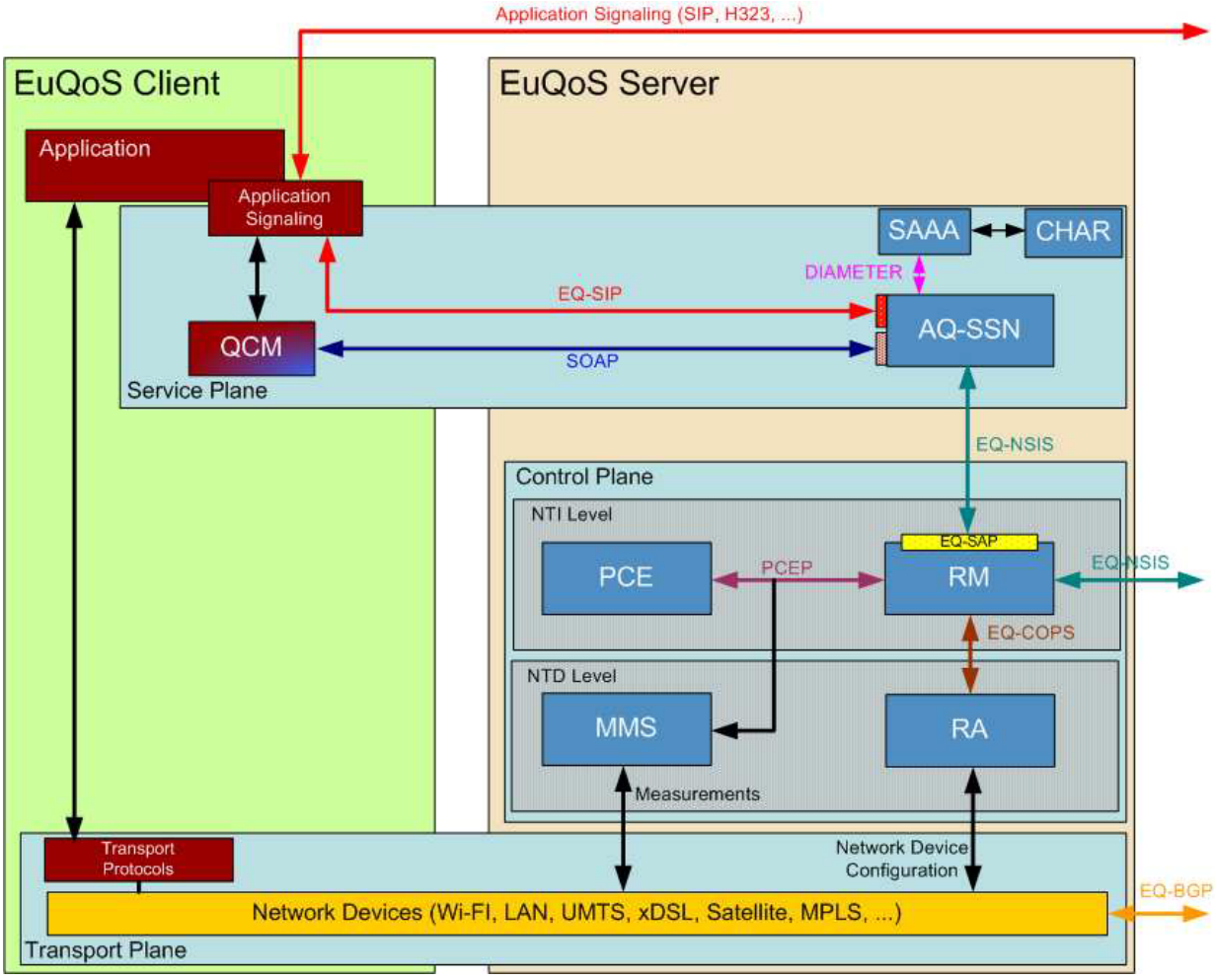


Figure 3.3: les composants d'EuQoS.

3.5.1 Les plans d'EuQoS

Le modèle de gestion de la QoS dans EuQoS a été divisé verticalement (Plan de Service, Plan de Contrôle et Plan de Transport) et horizontalement (suivant les domaines et les Systèmes Autonomes) (figure 3.4).

3.5.1.1 Le plan de service

Le plan de service offre les différentes interfaces nécessaires à l'accès aux services proposés par EuQoS (communication directe pour les applications totalement compatibles EuQoS, interface WEB ouverte, extension du protocole NSIS). Il permet de plus de déclencher le processus de réservation de ressources de bout-en-bout. De plus, ce niveau est aussi responsable de la sécurité, de l'authentification, de l'autorisation et de la tarification (SAAA). Par ailleurs, il fournit un mécanisme de filtrage des requêtes de QoS suivant le profil et les droits de l'utilisateur.

Conçu explicitement pour offrir la QoS pour n'importe quelle application, EuQoS n'impose pas l'utilisation d'un protocole spécifique pour la signalisation de QoS entre les applications distantes. Les applications désirant de communiquer entre eux sont alors libre de choisir un protocole standard (SIP, H323, ...) ou propriétaire pour localiser chacun des autres, et négocier, entre eux, les paramètres de trafic (codec, adresse IP, ...). Cette étape est transparente pour le système EuQoS. Une fois les applications distantes se mettent d'accord sur les paramètres de QoS, l'invocation de service EuQoS s'amorce. L'invocation de service se fait à travers deux interfaces : QCM (Quality Control Module) du côté client, et « QoS-on-demand servicel » du côté système EuQoS. QoS-on-demand servicel est une interface exposée par le plan de service d'EuQoS, qui permet au client d'établir, finir et/ou modifier une session EuQoS.

Les requêtes de QoS qui arrivent au système à travers l'interface QoS-on-demand servicel sont orientées au module AQ-SSN (Application Quality Service Signalling Negotiation). Ce dernier interroge le serveur SAAA en utilisant le protocole Diameter [RFC3588] pour voir si le client est authentifié et autorisé à demander ce service. Si la demande est acceptée le module AQ-SSN utilise l'interface EQ-SAP (EuQoS Service Access Point) (interface du plan contrôle avec le plan service) pour demander du plan de contrôle de déclencher le processus de réservation de ressources, et demande également de SAAA de commencer la comptabilisation du flot de la session. Après la fin de la session, SAAA envoie les enregistrements de comptabilisations au module CHAR (Charing) qui calcule la facture des sessions.

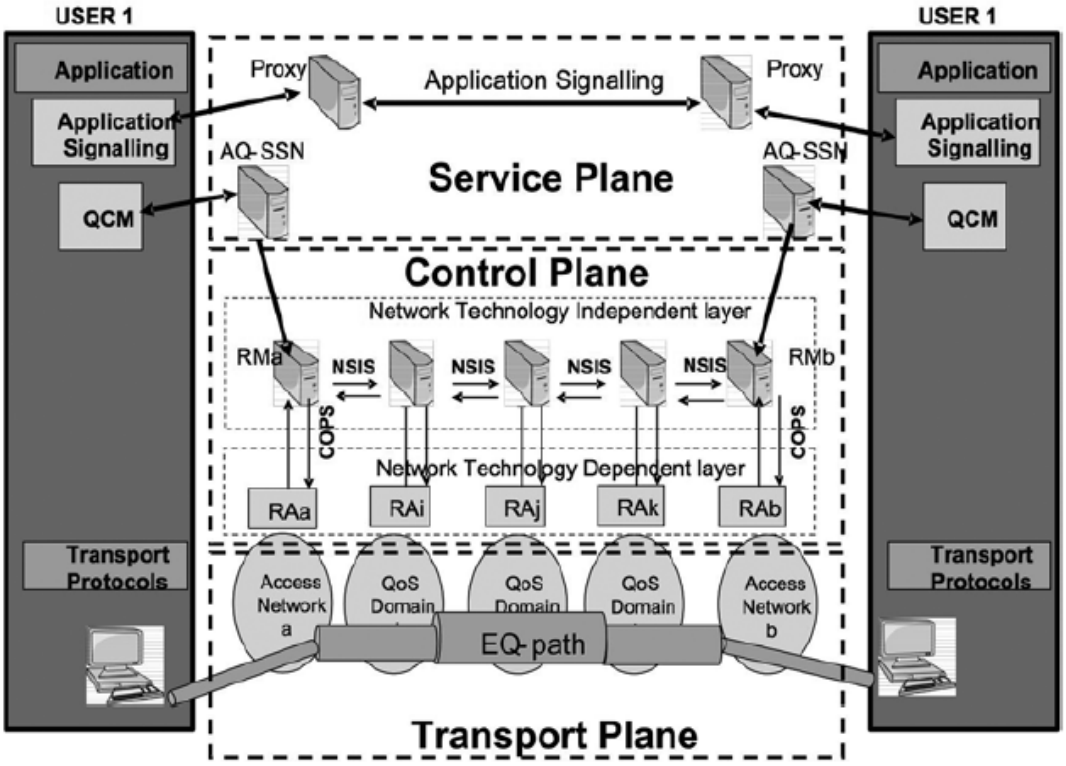


Figure 3.4: l'architecture d'EuQoS.

3.5.1.2 Le plan de contrôle

Le plan de contrôle contient toutes les protocoles et les fonctionnalités nécessaires pour la création, l'utilisation et le contrôle des chemins de bout-en-bout et les ressources y associées. Ces chemins sont appelés EQ-Paths.

3.5.1.2.1 Construction d'EQ-Path

Dans EuQoS, l'EQ-Path est sélectionné par l'utilisation du protocole de routage inter-domaine EQ-BGP, protocole modifié du BGP. EQ-BGP dispose d'un nouveau champ QoS_NLRI (QoS Network Layer Reachability Information) qui transporte des informations sur les paramètres de QoS, les paramètres de configuration nécessaire pour le processus de sélection de chemin et la table de routage par classe.

La figure 3.5 montre comment le NLRI est calculé et publié par EQ-BGP pour une CoS donnée. Soient Q_A, Q_B et Q_C les valeurs d'un paramètre donné de QoS (délai par exemple) dans les domaines A, B et C respectivement, et $Q_{A\to B}$, $Q_{B\to A}$ et $Q_{C\to B}$ les valeurs de QoS du même paramètre sur les liens inter-domaines. Ces valeurs sont configurées par le module TERO dans chaque domaine. Quand le domaine C annonce une nouvelle destination au A à travers B, le champ $NLRI_c$ correspondant à cette destination est mis à jour progressivement en utilisant une fonction de composition de QoS appropriée (c'et l'addition dans cet exemple). De cette manière, le domaine A peut alors déduire que la valeur correspondant à $NLRI_c$ est égal à $Q_C \oplus Q_{B\to C} \oplus Q_B \oplus Q_{A\to B}$; tel que l'opérateur $\oplus$ représente la fonction de composition générique de QoS.

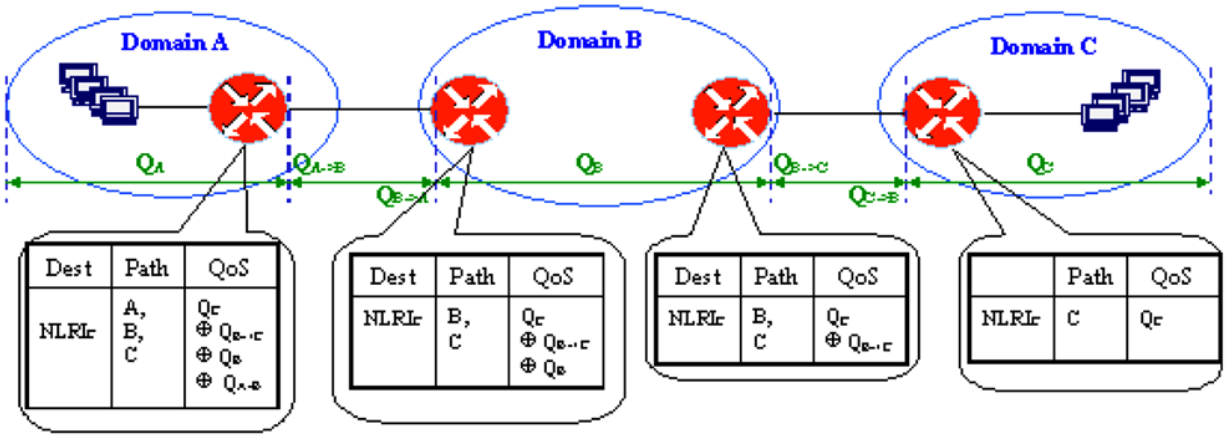


Figure 3.5: Opération d'EQ-BGP.

Le module TERO calcule les valeurs de paramètres de QoS périodiquement, pour la maintenance et l'optimisation, et aussi en réaction au changement de la topologie de réseau (par exemple, suite à la négociation d'un nouveau lien inter-domaine, p-SLS, avec l'adjacent).

3.5.1.2.2 Le provisionnement

Le rôle du protocole EQ-BGP se limite à déterminer les valeurs de paramètres de QoS sur les liens entre les différents domaines. L'étape suivante est la réservation, au long terme, de ressources propres à EQ-Path (un EQ-Path correspond à une seule e2e CoS). Cette étape s'appelle le « provisionnement ». EuQoS propose deux modèles de provisionnement : « loose model » et « hard model ».

Loose model

Dans le modèle loose, chaque domaine s'approvisionne des ressources de ses domaines adjacents, et configure ainsi, ses routeurs conformément au contrat de peering (p-SLS). Avant l'établissement d'une nouvelle connexion, un processus d'invocation est lancé, qui consiste à signaler et réserver (appel au contrôle d'admission) au long d'EQ-Path.

Le principal avantage du loose model est qu'il introduit un couplage minimal entre les domaines impliqués dans l'EQ-Path. Il n'exige que l'existence des accords entre les domaines pairs et la possibilité de mettre en place la signalisation EuQoS. L'inconvénient majeur du loose model consiste dans le nombre de messages de signalisation échangés lors d'une réservation/libération des ressources à chaque requête de QoS.

Hard model

La deuxième approche proposée dans EuQoS adopte la vision des opérateurs de télécommunication et est fondée sur le concept d'EQ-Link. Un EQ-Link est un lien virtuel entre deux routeurs de bordure, potentiellement dans des domaines non adjacents (mécanisme de tunnel). L'EQ-Link présente des caractéristiques de QoS bien définies entre ces deux nœuds : les EQ-Link sont associés à une CoS spécifique (et non pas à une connexion) pour laquelle les ressources sont réservées en avance. Ainsi, un EQ-Path peut être construit dans le processus de provisionnement par la demande d'établissement d'un EQ-Link entre deux domaines. Les EQ-Link qui traversent plusieurs domaines sont considérés comme des liens inter-domaines par les autres modules de l'architecture EuQoS. En pratique, les EQ-Links EuQoS sont établis en tant que tunnels, par exemple DiffServ MPLS-TE, et peuvent traverser plusieurs AS. La mise en place du provisionnement hard-model repose sur l'architecture Path Computation Element (PCE).

La figure 3.6 illustre l'établissement de connexion en hard model. Premièrement les domaines AS_1, AS_2 et AS_3 se coopèrent pour établir un EQ-Link entre les routeurs $R_{1.2}$ et $R_{4.1}$. Quand le hôte A veut ouvrir une connexion de classe CoS_x avec le hôte B, il utilise l'EQ-Link rétabli (ou bien provisionné) auparavant entre $R_{1.2}$ et $R_{4.1}$. Dans ce cas, le contrôle d'admission est effectué seulement à l'extrémité d'entrée d'EQ-Link (AS_1 dans l'exemple), au contraire du loose model.

Néanmoins, ce type de provisionnement nécessite une forte coopération entre les AS et sa mise en place est plus complexe. De plus, il n'est applicable que dans certains réseaux de cœurs, comme ceux qui implémentent des mécanismes de type DiffServ MPLS-TE.

En général, les EQ-Links ne sont pas établis de bout-en-bout (entre deux domaines d'accès), mais ils couvriront une portion de l'EQ-Path. Par conséquent, un EQ-Path peut être construit, de manière souple, de divers segments qui alternent les modèles loose et hard.

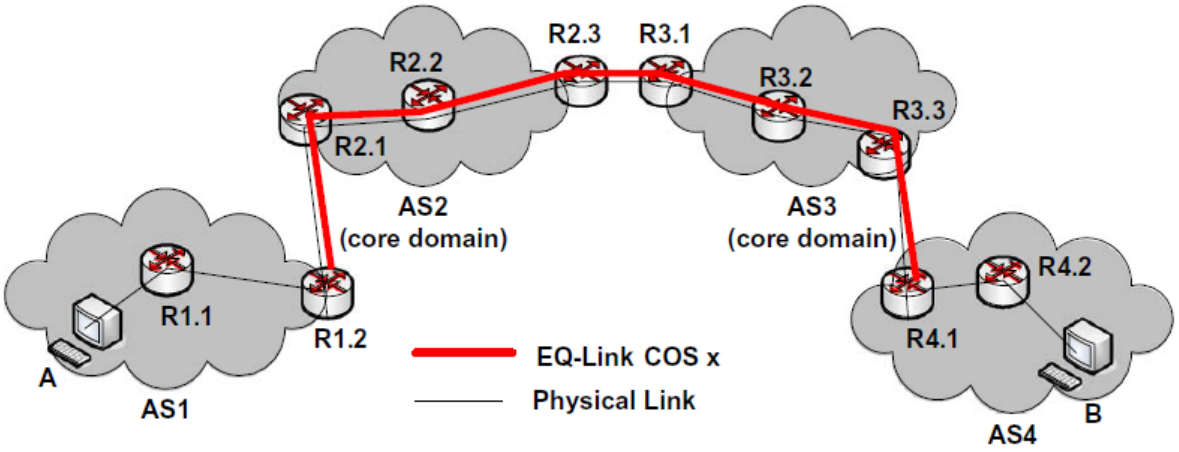


Figure 3.6: EQ-Link entre deux domaines.

3.5.1.2.3 L'utilisation d'EQ-Path (processus d'invocation)

Le processus d'invocation diffère suivant le modèle de provisionnement : loose ou hard.

Dans le cas de provisionnement loose, on suppose que les domaines aient coordonné, chacun avec ses voisins, pour s'approvisionner de ressources en concluant un contrat de peering. Le RM, se situant au plan de contrôle, reçoit les requêtes de QoS via l'interface EQ-SAP (EQ-Service Access Point) qui est exposée au plan de service, et fait les vérifications suivantes :

- si la requête provient du domaine local, le RM fait appel au processus de contrôle d'admission de bout-en-bout (end-to-end CAC) qui consiste à constater de l'existence d'un EQ-Path menant au domaine destinataire et correspondant à la classe (e2e CoS) de la requête ;
- détermine le lien inter-domaine et chemin intra-domaine à emprunter vers le domaine suivant, et effectue le contrôle d'admission de domaine (domain CAC) pour voir s'il a, à l'intérieur du domaine, suffisamment de ressources pour accepter la requête. Ce dernier contrôle est réalisé en interaction avec le RA (Resource Allocator) ;
- passe la requête au RM du domaine suivant au long du chemin EQ-Path qui, à son tour, répète les étapes précédentes, et ainsi jusqu'au domaine destinataire.

La figure 3.7 montre les interactions horizontales (entre les RM) et les interactions verticales (entre RM et RA) entre deux domaines afin de signaler une requête de QoS de bout-en-bout. Le protocole utilisé pour les interactions horizontales est EQ-NSIS, extension du NSIS, qui assure que les messages de signalisation parcourent le même EQ-Path que les données (passent par les mêmes domaines et mêmes routeurs inter-domine). Pour les interactions verticales le protocole utilisé est EQ-COPS, extension du protocole COPS.

Premièrement le RM1 reçoit, du module AQ-SSN, le message Reserve Commit décrivant la QoS demandée. Après avoir constaté de l'existence d'EQ-Path, par l'exécution d'e2e CAC, RM1 interroge le RA1. Si la requête peut être satisfaite, par l'exécution de Domain CAC, RA1 « bloque » les ressources requises et envoie le message OK au RM1. RM1 alors enchaine la requête au RM2 et envoie au RA1 un message de confirmation de réservation. Celui-ci alloue les ressources sur les équipements de réseau et puis répond par le message *Commit*. RM2 effectue le même travail que RM1 et lui répond par un message confirmation de réservation. Une fois RM1 reçoit les deux confirmations (de RA1 et RM2), il répond au module AS-SSN que la réservation est faite et la connexion peut être activée.

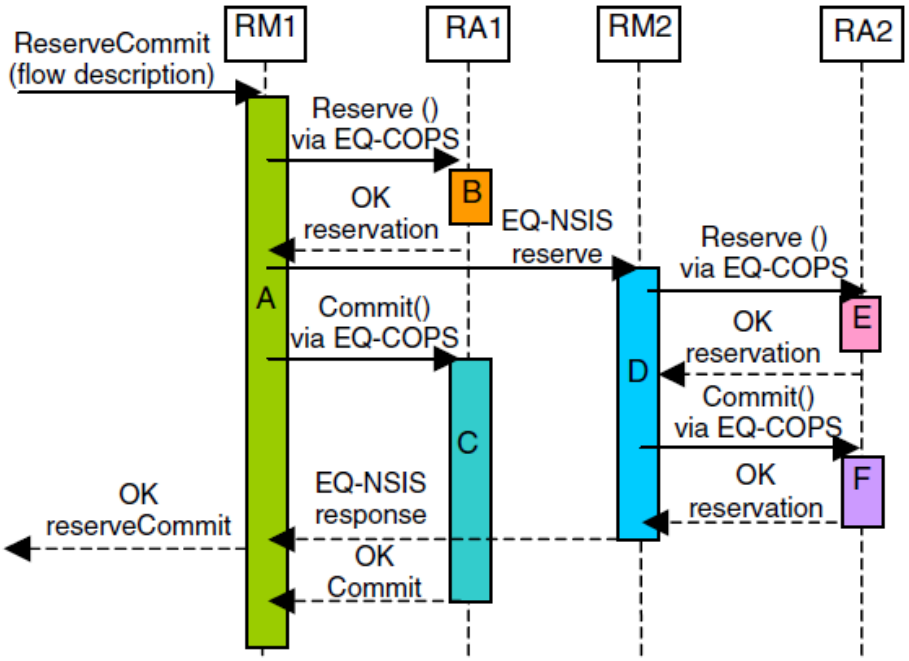


Figure 3.7: les interactions horizontales et verticales de signaliastion.

Dans le cas où le modèle provisionnement est hard model, le processus d'invocation se réduit à juste un contrôle d'admission de domaine (Domain CAC) à l'intérieur du domaine d'entrée d'EQ-Link.

3.6 Contrôle d'admission (CAC) dans EuQoS

Le contrôle d'admission est un élément clé de l'architecture EuQoS, compte tenu de l'hétérogénéité des technologies couvertes. EuQoS définit plusieurs CAC repartis sur les niveaux NTI et NTD :

- Un CAC intra-domaine, inter-domaine et de bout-en-bout (e2e) indépendant des technologies réseaux sous-jacents. Le processus de contrôle d'admission est réalisé sur les données présentes dans une base de données.
- Un CAC lié à la technologie sous-jacente, qui implémente le contrôle d'admission spécifique à chaque type de réseau. Le processus de contrôle d'admission met en œuvre alors des algorithmes dépendants de la technologie réseau. (LAN, xDSL, Wi-Fi, UMTS ou Satellite).

De plus, les modules de CAC manipulent des paramètres de QoS à plusieurs niveaux. Ces paramètres, sont spécifiés dans les SLS. Différents aspects des SLS ont été définis dans l'architecture proposée dans EuQoS :

- a-SLS : paramètres reçus par le RM, en particulier le premier sur l'EQ-Path;
- e-SLS : paramètres du SLS relatifs au chemin de bout-en-bout ;
- d-SLS : paramètres à QoS concernant le domaine géré par le RM (intra-domaine) ;
- i-SLS : paramètres liés à la partie inter-domaine ;
- r-SLS : paramètres passés au prochain RM.

Le CAC est divisé en sous-modules répartis de manière hiérarchique dans les composantes RM et RA. Ces modules sont le CAC de bout en bout (End-to-End CAC), le CAC à l'intérieur d'un domaine (Intra Domain CAC), le CAC inter-domaine (Inter Domain CAC) et le CAC de la technologie sous-jacente (UN CAC) (figure 3.8).

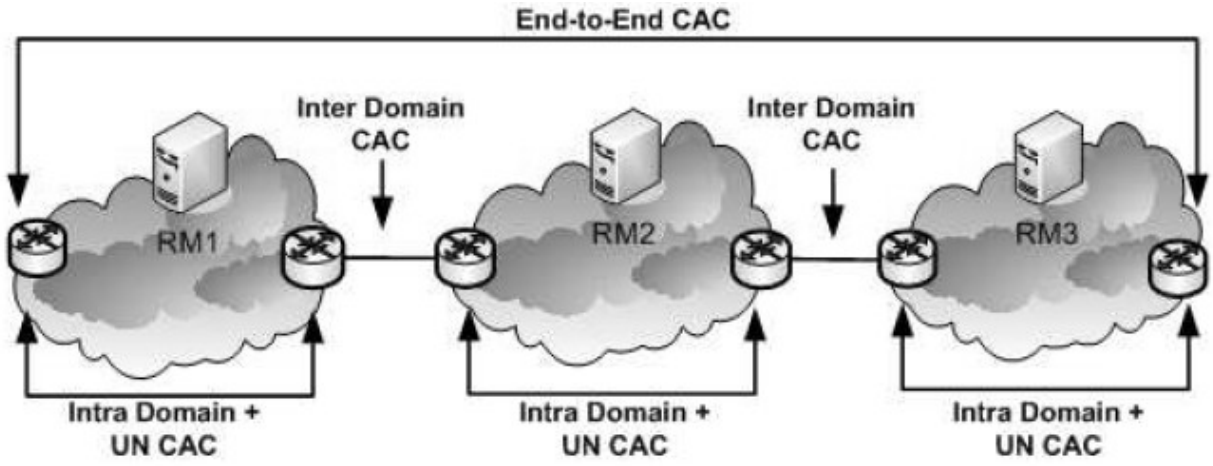


Figure 3.8: les différents CAC d'EuQoS.

3.6.1 CAC de bout en bout (End-to-end CAC)

Ce module prend en charge la vérification de l'existence d'un chemin de bout-en-bout (EQPath) correspondant à la CoS e2e requise. L'EQ-Path est examiné une seule fois pour trouver la séquence de domaines entre la source et la destination. Le module e2e CAC est sollicité seulement dans le RM du premier domaine sur le chemin de données.

3.6.2 CAC intra-domaine (Intra Domain CAC)

Même si un EQ-Path qui répond aux besoins en QoS a été trouvé, il est possible qu'au moment de la requête les ressources ne soient pas disponibles tout au long du chemin. Le CAC intra-domaine est spécifique à chaque domaine ; il est donc difficile de définir un mécanisme générique efficace applicable à toutes les technologies. La solution adoptée en EuQoS était de laisser l'implémentation de ce module à chaque administrateur de domaine qui pourra également tenir compte le type de réseau sous-jacent.

3.6.3 CAC inter-domaine (Inter Domain CAC)

Pour garantir la QoS de bout en bout, EuQoS défini un module spécifique qui prend en charge la vérification de la disponibilité des ressources sur les liens inter-domaines (sur chaque lien de peering dans loose model et sur chaque lien virtuel crée dans hard model).

3.7 Conclusion

L'hétérogénéité de la technologie des domaines pose un vrai problème devant l'assurance de la qualité de service de bout-en-bout dans un environnement multi-domaine.

Le système EuQoS résout bien ce problème et permet à l'opérateur d'offrir multiservice garantissant des objectifs de QoS de bout-en-bout sans se soucier des technologies sous-jacentes des autres domaines et ses manières de garantir la QoS.

Le système EuQoS affecte chaque nouveau flux à la classe qui lui correspond, en lui attribuant ainsi les ressources et le traitement adéquats. Que fera-t-il s'il ne peut pas déterminer la classe correspondante à un nouveau flux (pour des raisons de sécurité par exemple). Dans le reste du mémoire nous présenterons notre contribution à ce propos.

4 Contributions : Simulations et Solutions

Comme nous l'avons déjà expliqué, le système EuQoS vise à offrir la QoS dans un environnement multiservice, multi-domaine et multi-technologie. Il définit et gère des classes de services abstraites qui sont mappées dans chaque domaine en classes de service spécifiques à la technologie sous-jacente. Il agrège les flux de même nature en une seule classe et y attribue de ressources partagée entre ses trafics. Le problème avec ce système EuQoS est qu'il ne peut pas gérer un flux sans connaitre sa nature (CBR, VBR, élastique etc.) ; Autrement dit, il ne peut pas affecter un flux dont il ne connait pas initialement la nature à une classe donnée. Un bon exemple est le flux dans le cas d'une sécurisée où le client ne dévoile pas la nature du trafic.

Dans tous les cas, la méconnaissance du modèle de trafic empêche le système de l'affecter à une classe de service, et par conséquence, lui attribuer les ressources et les traitements appropriés.

Notre contribution directe, donc notre objectif, est de permettre au système d'offrir une garantie de service aux flux dont la nature est inconnue.

Théoriquement il y a deux possibilités : soit agréger le flux à nature inconnue en une des classes prédéfinies, soit définir un nouveau service qui gère les flux individuellement.

Dans ce chapitre, nous examinons d'abord la première possibilité pour vérifier l'effet d'agréger un flux de classe quelconque à une autre classe. Nous utilisons les classes de services et les algorithmes de contrôles d'admission définies par le projet AQUILA (prédécesseur d'EuQoS). [Bak1] [Bak2].

Nous présentons ci-dessous une brève explication sur l'architecture du projet AQUILA.

La deuxième partie du chapitre sera dédiée à notre proposition pour l'extension au projet EuQoS afin qu'il puisse traiter en plus les flux de nature inconnue.

4.1 Simulations

4.1.1 L'architecture AQUILA

AQUILA (Adaptative Resource Control for QoS Using an IP-based Layered Architecture) est un projet européen qui visait la conception et l'implémentation d'une architecture complète de QoS dans un environnement IP. Il introduit la définition de services réseaux, la spécification des règles de contrôle d'admission pour chaque service et la spécification des mécanismes de signalisation dynamique. Il ajoute à l'architecture DiffServ une couche de contrôle de ressource (RCL : Resource Control Layer) qui joue le rôle d'un « Bandwidth Broker » distribué.

Aquila définit les services : Premium CBR (PCBR), Premium VBR (PVBR), premium MultiMedia (PMM) et Premium Mission Critical (PMC), en plus de service standard Best Effort.

PCBR est conçu pour les applications qui nécessitent un débit constant, délai et taux de perte très petits comme les applications de la voie. Le trafic de PCBR est caractérisé par le débit crête (peak rate).
Le service PVBR est conçu pour les applications qui nécessitent un débit variable et les mêmes critères que le service PCBR comme les applications de la vidéo. Le trafic PVBR est caractérisé par le débit soutenu (Sustainable Rate) et le débit crête (Peak Rate).
Les services PMM et PMC sont conçus pour les trafics élastiques. PMM supporte la longue rafale (Burstiness) et caractérise le trafic par « Sustainable Rate » seulement. Le peak rate d'un trafic PMM

est égal au débit du lien de la communication. Le trafic PMC est caractérisé par le Sustainable Rate (SR) et Peak Rate (PR).

4.1.2 Spécifications des services Aquila

Avant de donner les spécifications de service, il est utile de définir les abréviations suivantes.

- PR = Peak Rate (bit/s).
- BSP = Bucket Size for PR (bytes).
- SR = Sustainable Rate (bit/s).
- BSS = Bucket Size for SR (bytes).
- M = Maximum Allowed Packet Size.
- m = Minimum Policed Unit.

4.1.2.1 Premium CBR

- QoS: délai ≤ 150 ms pour 99.99 %, $P_{loss} \leq 10^{-8}$.
- Style de réservation: p2p.
- Descripteur de trafic (TD) : Sigle Rate.

Parameter	*Minimum admitted*	*maximum admitted*	*Default*
PR	0	200 Kb/s	…
M	40 B	256 B	40 B
M	n.a	n.a.	256 B
BSP	n.a	n.a	256 B

Tableau 3 : TD du service PCBR.

4.1.2.2 Premium VBR

- QoS : délai ≤ 150 ms pour 99.99 %, $P_{loss} \leq 10^{-4}$.
- Style de réservation: p2p.
- Descripteur de trafic (TD) : Dual Token Buket.

Parameter	*minimum admitted*	*maximum admitted*	*default*
PR	0	1 Mb/s	…
SR	0	PR	PR
BSS	M	??	…
M	40 B	M	40 B
M	n.a	n.a.	512 B
BSP	n.a	n.a	1024 B

Tableau 4 : TD du service PVBR.

4.1.2.3 Premium MultiMedia

- QoS : $P_{loss} \leq 10^{-3}$ pour les paquets in-profile, pas de QoS pour les paquets out-of-profile.
- Style de réservation: p2p.
- Descripteur de trafic (TD) : Single Token Buket.

parameter	*minimum admitted*	*maximum admitted*	*default*
SR	0	250 Kb/s	100 Kb/s
BSS	M	??	??
m	40 B	M	40 B
M	n.a	n.a.	512 B

Tableau 5 : TD du service PMM.

4.1.2.4 *Premium Mission Critical*

- QoS : $P_{loss} \leq 10^{-6}$ pour les paquets in-profile, pas de QoS pour les paquets out-of-profile.
- Style de réservation: p2a.
- Descripteur de trafic (TD) : Double Token Buket.

parameter	*Minimum admitted*	*maximum admitted*	*default*
PR	0	50 Kb/s	…
SR	0	5 Kb/s	PR
BSS	M	10,000 Bytes	
m	40 B	M	40 B
M	n.a	n.a.	512 B
BSP	n.a	n.a	1024 B

Tableau 6 : TD du service PMC.

4.1.3 Contrôle d'Admission (CA)

Aquila définit 05 classes de trafic, TCL_{1-5}, où chaque classe de trafic correspond à un service. A chaque classe de trafic est associé un algorithme de contrôle d'admission dont le rôle est de garantir la QoS aux applications. Il limite l'accès au réseau de telle sorte que les objectifs de QoS restent respectés pour tous les flux en cours.

Le développement et l'implémentation d'un algorithme de contrôle d'admission nécessitent :

- Le descripteur de trafic (TD) : utilisé par l'utilisateur pour décrire le trafic qu'il va injecter dans le réseau. Les modèles utilisés sont : Sigle Rate, Dual Token Bucket, etc.
- Les paramètres de QoS : comme le taux de perte de paquet, le délai;
- Le modèle de système caractérisant la bande passante disponible et le buffer dédié.

Dans notre simulation et pour orienter un flux vers une autre classe, et par conséquence lui y appliquer son contrôle d'admission, il faut adapter ses paramètres de TD afin qu'ils correspondent au TD des flux de la nouvelle classe.

4.1.3.1 *Classe de trafic TCL_1*

La classe TCL_1 est conçue pour gérer les flux CBR. Le trafic correspond à ce flux est modélisé par Sigle Token Bucket (TB).

Un nouveau flux, défini par le peak rate PR_{new} et la taille du seau BSP (Token Buket Size), est accepté si la condition suivante est vérifiée :

$$PR_{new} + \sum_{k=1}^{N} PR_k \leq \rho R$$

Où N est le nombre de flux déjà acceptés dans cette classe, R est la bande passante dédiée à la classe et ρ ($\rho < 1$) est la charge admissible maximale. La valeur ρ est obtenu par l'analyse du système de file d'attente M/D/1/B où elle représente l'utilisation maximale du système satisfaisant le taux de perte de paquets (P_{loss}) requis pour cette classe. La valeur de ρ est alors donnée par la relation :

$$\rho = \frac{2B}{2B - \ln(P_{loss})}$$

4.1.3.2 Classe de trafic TCL_2

La classe TCL_2 est conçue pour les flux VBR. Le trafic correspondant à un flux VBR est modélisé par double TB. Pour répondre au besoin de plus court délai, on utilise le schéma de multiplexage de flux REM (Rate Envelope Multiplexing) qui consiste à agréger, au niveau de routeur, les flux de même classe dans un buffer dimensionné juste pour absorber le confit dû à l'arrivé simultané de paquets. Le contrôle d'admission introduit la notion de bande passante effective qui représente la quantité de bande passante requise pour servir un flux donné en respectant les paramètres de QoS et en prenant en considération le gain de multiplexage.

Un nouveau flux, dont la bande passante effective est *Eff(new),* est accepté si la condition suivante est vérifiée :

$$Eff(new) + \sum_{k=1}^{N} Eff(k) \leq R$$

Cette formule est vraie à condition que le taux de perte soit donné par

$$P_{loss} = exp\left\{-2B\left(\frac{B}{N} - 1 + \frac{D}{N}\right)\right\}$$

Où N est le nombre de flux, et $D=R/PR_{min}$.

La méthode de calcul de bande passante effective dépend du schéma de multiplexage utilisé (c'est REM dans ce cas). Dans la classe TCL_2, *Eff(.)* est calculée par la formule :

$$Eff(.) = \begin{cases} a.AR\left(1 + 3z\left(1 - \frac{AR}{PR}\right)\right) \; si \; 3z \leq \min\left(3, \frac{PR}{AR}\right) \\ a.AR\left(1 + 3z^2\left(1 - \frac{AR}{PR}\right)\right) \; si \; 3 \leq 3z^2 \leq PR/AR \\ a.PR \; autrement \end{cases}$$

Où

$$a = 1 - \frac{log_{10}P_{loss}}{50} \; et \; z = -\frac{2log_{10}P_{loss}}{R/PR}$$

4.1.3.3 Classe de trafic TCL_3

La classe TCL_3 est conçue pour gérer les flux TCP. Le trafic correspondant est caractérisé par les deux paramètres de TB : SR et BSS. L'objectif de l'application d'un contrôle d'admission sur les flux TCP est de garantir un minimum de bande passante pour les flux en cours. Doter la classe d'un buffer de grande capacité cause un long délai de transfert paquet. Et, par contre, le buffer de petite capacité cause la hausse du taux de perte de paquet.

Un nouveau flux est accepté si les conditions suivantes sont vérifiées :

$$SR_{new} + \sum_{k=1}^{N} SR_k \leq R$$

Et

$$(N+1)M < B$$

Où N est le nombre de flux déjà acceptés, M la taille maximale des paquets de la classe TCL_3 et SR_{new} le débit soutenu du nouveau flux.

Dans cette classe le mécanisme Random Early Detection (RED) est utilisé pour assurer la stabilité de la file d'attente. La stabilité de la file d'attente signifie que l'amplitude de l'oscillation de la moyenne de la taille de file est minimale et l'oscillation reste autour de la valeur : $(maxth - minth)/2$. Si la moyenne de la taille de file est inférieure de Minth aucun paquet n'est détruit ; et en revanche tous les paquets au-delà de Maxth sont détruits. Si la moyenne est entre Minth et Maxth les paquets reçus sont détruits avec une probabilité déterminée. Les classes TCL_3 et TCL_4 utilisent le Weighted RED (WRED) qui se compose de deux RED : un pour les trafics in-profile et l'autre pour les trafics out-of-profile. Un code point se situant dans l'entête IP est associé à chaque RED.

4.1.3.4 Classe de trafic TCL_4

La classe TCL_4 est conçue pour les flux TCP de courte durée et nécessitant une bande passante limitée. Le trafic appartenant à la classe TCL_4 est modélisé par double TB. Le contrôle d'admission utilise, lui aussi, la notion de la bande passante effective avec le multiplexage REM.

Un nouveau trafic, caractérisé par les paramètres PR, SR et BSS, est accepté si la condition suivante est vérifiée :

$$Eff(new) + \sum_{k=1}^{N} Eff(k) \leq R$$

Où

$$Eff(.) = max\left\{SR, \frac{PR * T}{B/R + T}\right\} \; et \; T = \frac{BSS}{PR - SR}$$

4.1.4 Simulations

Pour voir l'effet de l'agrégation d'un flux d'une classe quelconque en une autre classe, nous utilisons le simulateur réseau NS2, la version 2.34. Ce choix est justifié par la crédibilité connue à NS2 et aussi à cause de sa gratuité. Nous l'avons installé et exécute sous la distribution linux Fedora 12. A chaque simulation, on considère une seule classe de service dotée d'un algorithme de contrôle d'admission et dimensionnée de ressources limitées (capacité transmission et buffer). Deux flux sont générés à

chaque fois : un correspond à la classe considérée, et l'autre à une autre classe. Celle-ci représente, dans notre proposition, le flux qu'on ne connait pas sa nature et qu'on veut l'agréger à une classe bien connue. De plus, comme le contrôle d'admission a besoin des paramètres de TD des flux pour prendre la décision, il est nécessaire de transformer le TD du flux étranger en TD de la classe concernée par la simulation. Chaque simulation est répétée 10 fois et un test statistique (t_test) pour la validation des résultats est effectué avec un niveau de risque égal à 0.05.

4.1.4.1 Agrégation de trafic TCL_2 en TCL_1

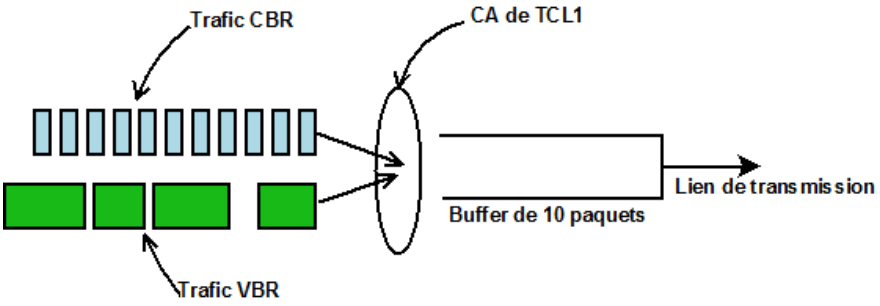


Figure 4.1 : agrégation de trafic TCL_2 en TCL_1.

Dans ce cas, les paramètres de QoS assurées par la classe sont : délai < 150 ms, variation de délai < 20 ms et taux de perte paquet < 10^{-2}.

En mettant la taille de buffer B à 10 la charge admissible ρ de la classe sera :

$$\rho = \frac{2B}{2B - \ln(P_{loss})} = 0.81$$

La décision de CA de TCL_1 est basée sur le paramètre PR alors que les flux de TCL_2 sont décrits par les paramètres PR, SR et BSS. On ignore donc les paramètres SR et BSS malgré leur importance pour décrire la nature de flux.

Dans cette simulation, et la simulation suivante, on attribue au trafic VBR 90% de la limite de transmission (égale à $\rho * capacité\ de\ transmission\ R$) et 10% au trafic CBR. Les valeurs données à R sont 10, 20, 50 et 100 Mb successivement.

A chaque simulation, nous muserons :

- le taux de perte (Loss ratio),
- le délai maximal (Max delay) en ms,
- la standard déviation (l'écart type) de délai pour chaque flux.

Les résultats sont présentés dans le tableau suivant :

	CBR1			VBR2		
Rate (Mb)	Loss ratio	Max delay (ms)	Standard deviation	Loss ratio	Max delay (ms)	Standard deviation
10	0,05222	10	0.0013	0.051967	23	0.0014
20	0,051739	5	0.0006	0.051726	12	0.0007
50	0,057158	2	0.0002	0.057086	5	0.0002
100	0,051938	1	0.0001	0.051090	2	0.0001

Tableau 7 : effet de l'agrégation de trafic TCL_2 en TCL_1.

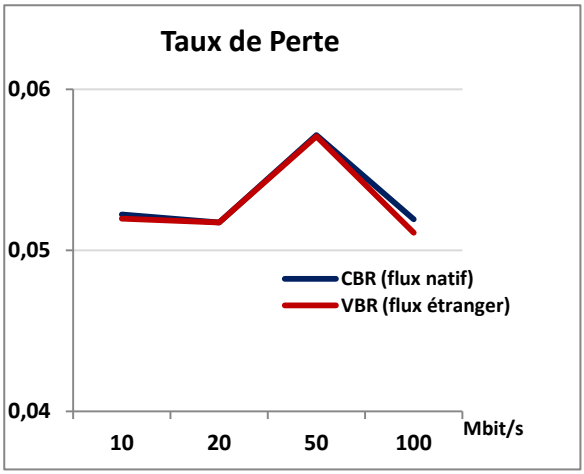


Figure 4.2: Taux de perte (flux TCL_2 en TCL_1).

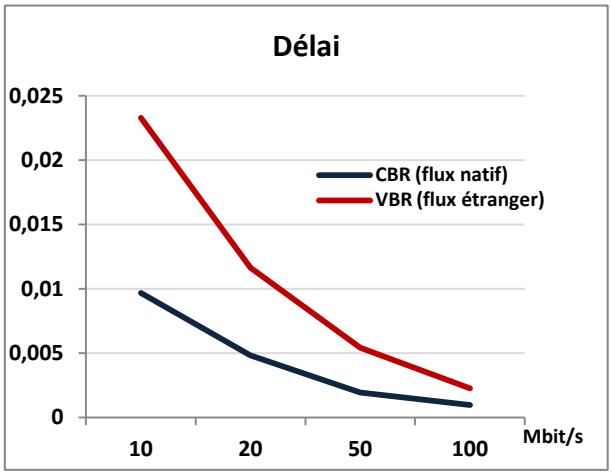


Figure 4.3: Délai (flux TCL_2 en TCL_1).

Le résultat montre que le taux de perte est à peu près fixé à une valeur plus élevée que prévue (0.05 au lieu de 0.01). Ce résultat est inacceptable complètement dans pour les applications CBR (comme la voie IP).

On constate que le délai maximal du flux CBR diminue de façon inversement proportionnelle que le débit de lien. Idem pour la standard déviation (0.0013 pour 10 Mb, 0.0006 pour 20 Mb, …).

Quant au flux VBR, l'impact est très impressionnant ! Le délai est très long par rapport au flux CBR. Mais il diminue également de façon inversement proportionnelle avec le débit du lien (23ms pour 10 Mb, 12ms pour 20 Mb, …). Idem pour la standard déviation diminue également de la même façon (0.0014 pour 10 Mb, 0.0007 pour 20 Mb, …).

4.1.4.2 Agrégation de trafic TCL_1 en TCL_2

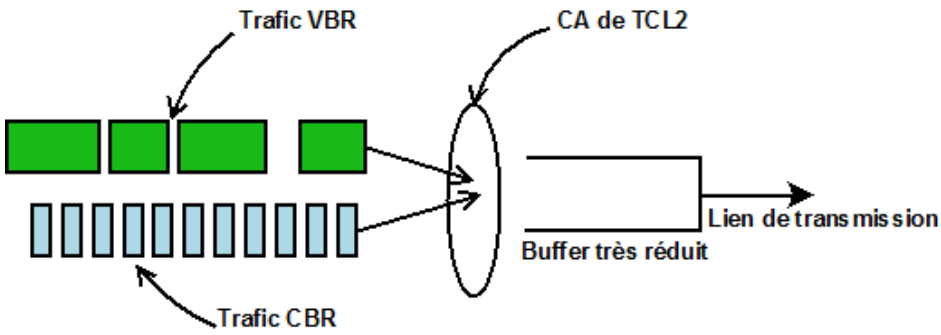


Figure 4.4: agrégation de trafic TCL_1 en TCL_2.

Le CA de TCL_2 est basé sur la notion de bande passante effective et a besoin de deux paramètres : PR et SR. Ces deux paramètres se trouvent dans le TD de flux TCL_2. Le TD de flux TCL_1 ne contient que le PR parce qu'il est conçu nativement pour un CA basé sur le PR (par contre au CA basé sur la bande passante effective). Nous considérons le PR de flux TCL_1 comme PR et SR à la fois afin de calculer la bande passante effective.

On note que la considération de SR comme PR conduit à une surallocation de débit de lien dédié à la classe car la bande passante effective dans ce cas est plus grande que le PR qui représente le maximum de débit utilisable par le flux.

La taille de buffer est déterminée par la formule vue auparavant, $P_{loss} = exp\left\{-2B\left(\frac{B}{N} - 1 + \frac{D}{N}\right)\right\}$, résultant du schéma de multiplexage de flux REM. en mettant N = 2, D=R/PR_{min} =10 (le trafic CBR prend toujours 10% de R) et P_{loss}=0.01, B devient 2. Ce qui signifie un très court délai de transfert, et permet d'ignorer le calcul de délai et la standard déviation dans la simulation.

Les deux flux, VBR et CBR, partagent le débit de lien R et le buffer B dédiés à la classe TCL_2 sous la condition que la somme de leurs bandes passantes effectives soit inférieure de R. A chaque simulation nous changeons R et, par conséquence, les débits du flux VBR et du flux CBR proportionnellement (90% pour le flux VBR et 10% pour le flux CBR), et nous traçons le taux de perte de paquet dans le tableau suivant :

	Loss Ratio	
Rate (Mb)	VBR	CBR
10,4	0	0,2
20,83	0	0,2
52	0	0,2
104	0	0,2
156	0	0,22
208	0	0,22

Tableau 8: effet de l'agrégation de trafic TCL_1 en TCL_2.

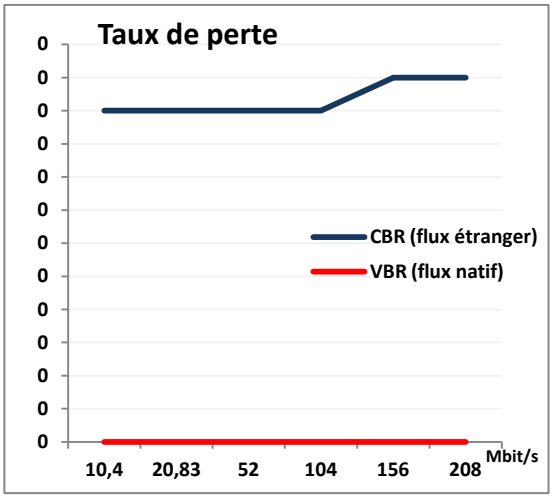


Figure 4.5: Taux de perte (flux TCL_1 en TCL_2).

Le résultat montre que pour toutes les valeurs données au débit de lien (de 10.4 Mb jusqu'à 208 Mb) le taux de perte paquet de flux VBR est nul, et celle de flux CBR est 0.2 (au lieu de 0.01 exigé). On constate alors que l'agrégation n'a aucun effet sur le flux VBR mais a un effet drastique sur le flux CBR.

4.1.4.3 Agrégation de trafic TCL_2 en TCL_3

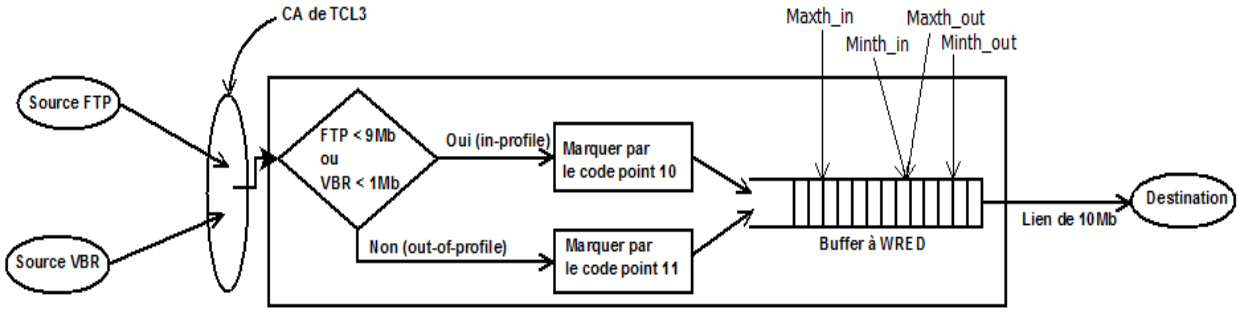


Figure 4.6: agrégation de trafic TCL_2 en TCL_3.

Le CA de TCL_3 garantie pour chaque flux une bande passante minimale égale à son SR. L'adaptation de TD de flux TCL_2 au CA de TCL_3 se fait par l'élimination de PR et l'utilisation de SR seulement. La condition d'acceptation d'un nouveau flux est : $SR_{new} + \sum_{k=1}^{N} SR_k \leq R$.

Le générateur de trafic FTP, disposé par le simulateur NS2, est utilisé pour simuler le flux TCL_3, et le générateur de trafic exponentiel ON/OFF pour simuler le flux VBR. Les deux flux partagent un débit de 10 Mb dédié à la classe. Nous définissons une règle de contrôle de dédit (policing) de telle sorte que les paquets qui dépassent la limite de son flux (9 Mb pour le flux FTP et 1 Mb pour le flux VBR) sont considérés out-of-profile, et on leur attribue le code point 11, et les paquets conformant leurs limites sont considérés in-profile, et on leur attribue le code point 10. Deux RED sont définis : un pour les paquets in-profile, marqués par le code point 10, et un autre pour les paquets out-of-profile, marqués par le code point 11. Notre teste consiste à changer les bornes de RED (Maxth et Minth) et tracer à chaque fois les valeurs de taux de perte, délai maximal, délai moyen et la standard déviation. Le tableau suivant résume les résultats :

$Minth_{out}$-$Maxth_{out}$, $Minth_{in}$-$Maxth_{in}$	FTP				VBR			
	Loss ratio	Dmax (ms)	Davg (ms)	Sdev	Loss ratio	Dmax (ms)	Davg (ms)	Sdev
2-5, 5-10	0,001	16	12	0,003	0,1	16	13	0,003
5-10, 10-20	0,00001	16	14	0,002	0,001	16	15	0,001
5-15, 15-25	0,000006	16	14	0,002	0,0001	16	15	0,001
5-20, 20-30	0,000006	16	14	0,002	0,0002	16	15	0,001
5-25, 25-35	0,000008	16	14	0,002	0,0003	16	15	0,001
5-30, 30-40	0,000002	16	14	0,002	0,0001	16	15	0,001
5-35, 35-45	0,000002	16	14	0,002	0,00005	16	15	0,001

Tableau 9: effet de l'agrégation de trafic TCL_2 en TCL_3.

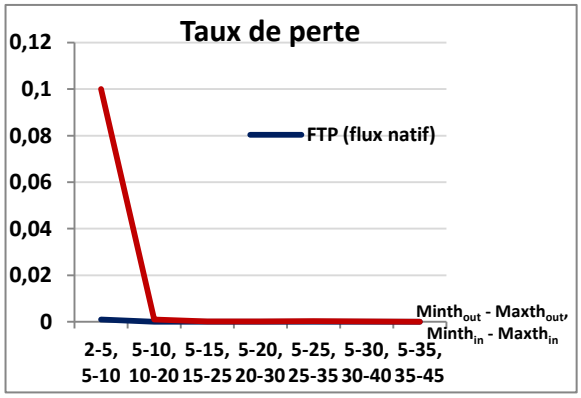


Figure 4.7: Taux de perte (flux TCL2 en TCL3).

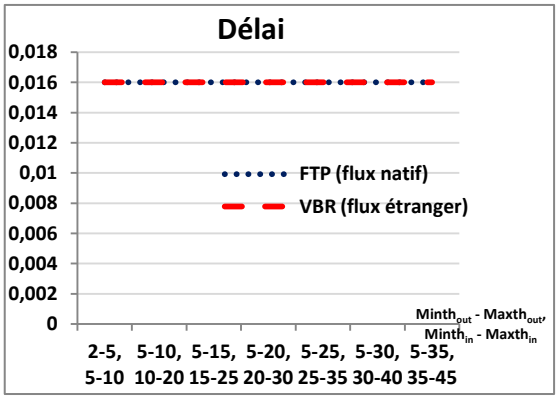


Figure 4.8: Délai (flux TCL2 en TCL3).

Les résultats montrent que le taux de perte de flux FTP est en diminution de petit à petit (de 0.001 à ≈ 0), et le délai est fixé à une valeur de 16 ms qui est acceptable pour ce type de flux. Quant au flux VBR, son taux de perte vaut premièrement très élevé mais continue de diminuer vers le 0. Il subit aussi le même délai que le flux FTP mais cette fois sa valeur peut influencer sur le flux.

4.1.4.4 Agrégation de trafic TCL_1 en TCL_3

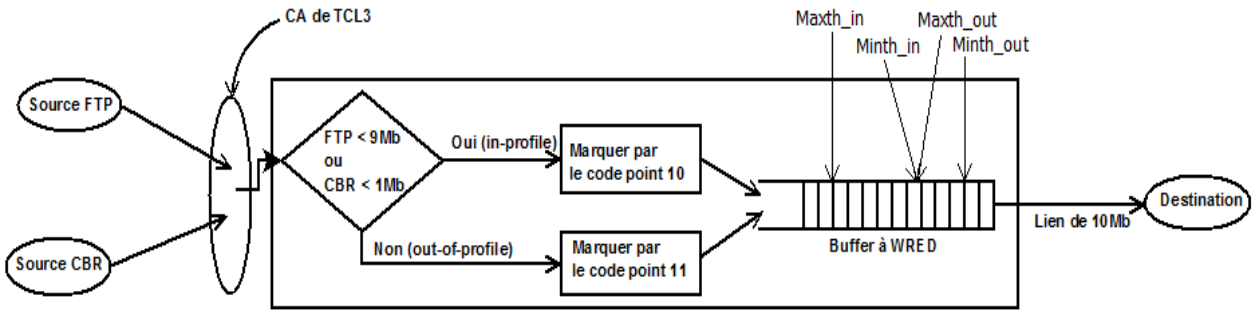


Figure 4.9: agrégation de trafic TCL_1 en TCL_3.

On répète la simulation précédente en remplaçant le flux VBR par le flux CBR. Cette fois l'adaptation de TD de flux CBR au CA de classe TCL_3 se fait par la considération de son PR comme SR, ce qui signifie que le CA va garantir au flux CBR une bande passante équivalente à son PR. Les résultats sont résumés dans le tableau suivant.

$Minth_{out}$-$Maxth_{out}$, $Minth_{in}$-$Maxth_{in}$	FTP				CBR			
	Loss ratio	Dmax (ms)	Davg (ms)	Sdev	Loss ratio	Dmax (ms)	Davg (ms)	Sdev
2-5, 5-10	0,03	14	10	0,004	0,024	14	10	0,004
5-10, 10-20	0,002	14	11	0,004	0,002	14	10	0,004
5-15, 15-25	0,001	14	12	0,003	0,001	14	12	0,003
5-20, 20-30	0,0003	14	13	0,002	0,0003	14	13	0,002
5-25, 25-35	0	14	14	0,0001	0	14	13	0,0004
5-30, 30-40	0	14	14	0,0001	0	14	13	0,0004
5-35, 35-45	0	14	14	0,0001	0	14	13	0,0004

Tableau 10: effet de l'agrégation de flux TCL_1 en TCL_3.

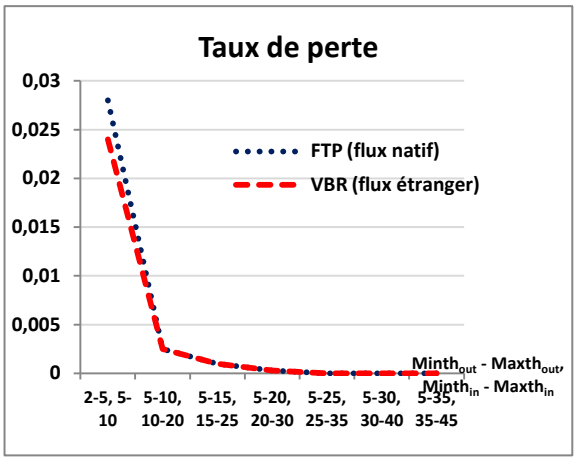


Figure 4.10: Taux de perte (flux TCL1 en TCL3).

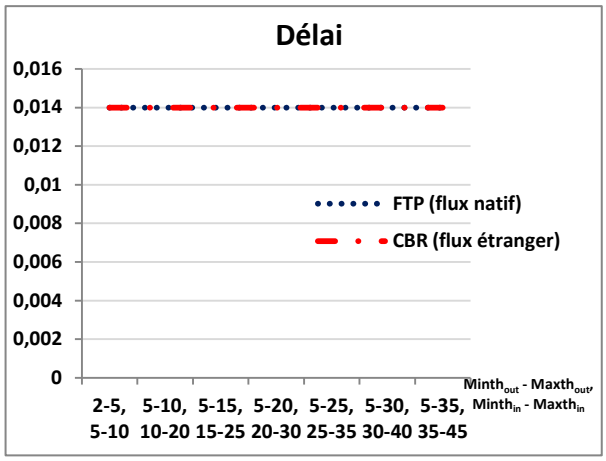


Figure 4.11: Délai (flux TCL1 en TCL3).

Les résultats montrent que l'effet de l'agrégation sur les paramètres de QoS est semblable pour les deux flux. Pour des petites valeurs de bornes de WRED ($Minth_{out}$, $Maxth_{out}$, $Minth_{in}$ et $Maxth_{in}$) le taux de perte est plus ou moins grand (0.03 et 0.024), mais en augmentant les bornes il diminue vers la valeur 0. De plus, les flux subissent le même délai, égale à 14 ms, qui peut introduire un effet indésirable sur le flux CBR.

4.1.5 Conclusion

Une classe de service est conçue pour multiplexer un ensemble de flux ayant la même nature, qui se traduit par le TD, et nécessitant les mêmes paramètres de QoS. Le contrôle d'admission (CA) assure l'utilisation des ressources de la classe par les flux sans altérer la QoS. Notre objectif dans cette première contribution était de voir l'effet de l'agrégation d'un flux d'une classe quelconque en une autre classe. Ceci devait nous permettre de constater les effets sur l'opérabilité de l'hypothèse qui consiste à orienter un flux à nature inconnue (pour des raisons de sécurité par exemple) vers une classe proche en termes de paramètres de QoS.

La première difficulté vient de l'impossibilité d'appliquer le CA sans connaître tous les éléments du descripteur de trafic TD. L'élimination ou le remplacement d'un élément par un autre (SR par PR par exemple) entraîne à une mauvaise décision par le CA.

Les résultats de simulation montrent bien que malgré que les classes de flux assurent les mêmes paramètres de QoS, l'agrégation d'un flux d'une classe dans une autre classe entraîne la dégradation de la QoS. On a constaté aussi que même si la classe assure une bande passante minimale pour ses flux, l'agrégation entraîne un délai influent négativement sur les flux des applications temps réel.

Ces résultats nous ont poussés à chercher une autre solution plus adéquate. Notre proposition porte sur la gestion de chaque flux à nature inconnu individuellement. Dans la deuxième partie du présent chapitre, nous présenterons notre solution qui permet au système EuQoS d'offrir le service par flux dans un environnement multi-domaine et hétérogène. C'est par conséquent comme une extension au système EuQos.

4.2 Extension du projet EuQoS : QoS par Flux

Après avoir montré l'effet indésirable de l'agrégation d'un flux à une classe qui n'est pas conçue pour lui, nous proposons dans cette partie l'extension de système EuQoS en lui ajoutant la capacité de gérer les flux individuellement. L'objectif de cette extension est de permettre au système d'offrir une garantie de service aux trafics de nature inconnue.

La solution proposée consiste à gérer chacun des flux de nature inconnu séparément des autres en lui attribuant de ressources propres. Le service proposé est une sorte de ligne virtuelle à QoS de bout-en-bout.

Dans le contexte du système EuQoS, le problème de scalabilité, qui peut s'imposer, est minimal dans notre cas car la solution proposée ne concerne qu'une partie particulière du trafic (seulement les flux de nature inconnue). De plus, l'utilisation d'un mécanisme d'agrégation de trafic de service garantie permet de minimiser d'avantage le problème de scalabilité.

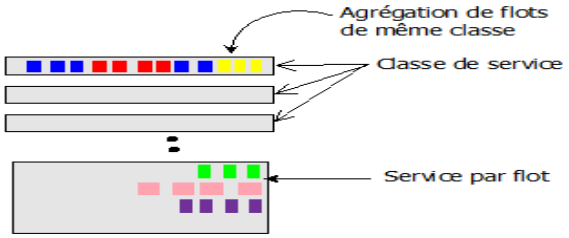


Figure 4.12: service orienté classe et service orienté flux.

Notre proposition consiste à ajouter un nouveau service qui traitera spécifiquement les flux de nature inconnue. Nous l'appellerons par la suite « Service par flux ».

4.3 Specifications:

4.3.1 Le provisionnement

Chaque domaine s'approvisionne de ressources nécessaires à chaque service en y réservant de la bande passante, de l'espace mémoire pour les buffers, etc.

EuQoS propose deux modèles de provisionnement: loose et hard. Dans le modèle loose, chaque domaine réserve de ressources de ses adjacents conformément au contrat de peering. Alors que dans le modèle hard un lien inter-domaine virtuel, EQ-Link, est établi entre deux nœuds et peut s'étaler sur plusieurs domaines. Le modèle hard, reposant sur l'architecture DiffServ MPLS TE, n'est pas adéquat pour le service par flux parce qu'on ne peut pas établir d'avance une liaison pour transporter un trafic inconnu. Même si la technologie MPLS permet d'affecter un flux donné à un chemin (LPS) propre, elle n'est pas conçue pour gérer les trafics par flux comme fait le paradigme IntServ dans le réseau IP.

Dans notre service nous utilisons le modèle loose qui se réduit à la mise en œuvre du contrat de peering (p-SLS) et laisse le grand travail à la signalisation.

4.3.2 Echange des informations inter-domaine

Le protocole EQ-BGP est modifié par l'ajout d'un nouveau champ pour transporter les informations sur l'état de ressources consacrées à ce service. Pour le moment, l'information qui concerne notre service est la bande passante disponible. La figure 4.6 montre la fonction ajoutée à l'EQ-BGP qui consiste à transporter la valeur de la bande passante disponible sur le chemin menant au domaine C. Dans cet exemple, sur chaque routeur une bande passante est allouée au service par flux : BR1 sur R1, BR2 sur R2, et ainsi de suite. La fonction de composition de paramètre de QoS utilisée est MIN qui retourne le minimum des bandes passantes disponibles au long du chemin inter-domaine menant au domaine C.

Le RM utilise ces informations pour effectuer le contrôle d'admission de bout-en-bout, *end-to-end CAC*, qui consiste à constater de l'existence d'un chemin de bout-en-bout dont la bande passante disponible est plus grand que la bande passante demandée.

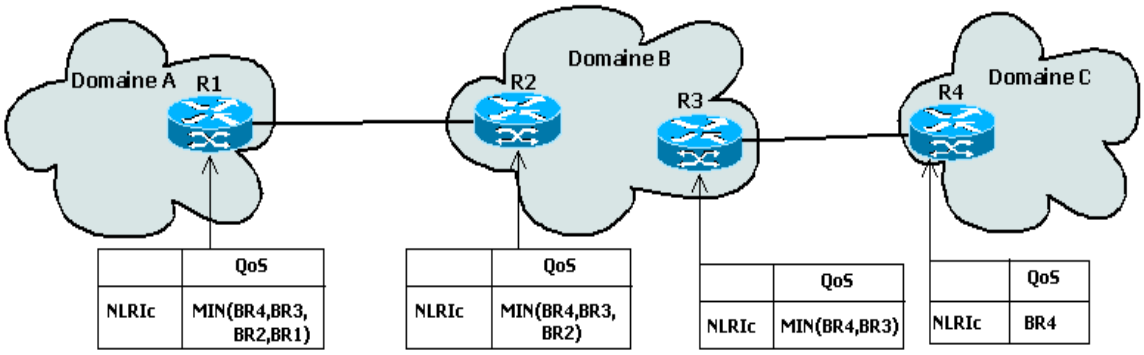


Figure 4.13: fonctionnement d'EQ-BGP modifié.

4.3.3 La signalisation

Le client signale au système que le trafic qu'il va générer ne doit pas être classé dans une aucune des classes prédéfinies. Il envoie également le descripteur du trafic en spécifiant les paramètres de Token Bucket : le débit d'arrivée de jetons (r), la taille de buffer (b) et le débit crête (p), et la taille maximale d'un paquet (M). La bande passante à réserver (R) doit vérifier l'équation :

$$\begin{cases} d_{e2e} = \dfrac{(b-M)(p-R)}{R(p-r)} + \dfrac{(M+C_{tot})}{R} + D_{tot} \;\; si\; p > R \geq r \;(1) \\ d_{e2e} = \dfrac{(M+C_{tot})}{R} + D_{tot} \;\; si\; R \geq p \geq r \;(2) \end{cases}$$

C_{tot} et D_{tot} sont les délais de bout-en-bout dus à la transmission de paquet et au traitement (identification de flux, recherche de l'interface de sortie, etc.) dans chaque routeur respectivement; et d_{e2e} est le maximum de délai de bout-en-bout à garantir. Le service par flux assure une communication sans perte de paquet.

Suivant la disponibilité, ou non, de ces informations (d_{e2e}, C_{tot} et D_{tot}), on distingue deux types de réservation : réservation déclenchée par l'émetteur et réservation déclenchée par le récepteur. La réservation déclenchée par l'émetteur a lieu dans le cas où l'émetteur sait d'avance la valeur de d_{e2e} à garantir et le RM, qui est lui associé, dispose les délais C_{tot} et D_{tot} (échangés par le protocole EQ-BGP modifié, par exemple). Elle permet de signalisation rapide.

La réservation déclenchée par le récepteur porte sur l'hypothèse que c'est le récepteur qui détermine la valeur de d_{e2e}, et par conséquent la valeur de R à réserver. C'est la démarche du protocole RSVP dans l'architecture IntServ. De plus, les valeurs de C_{tot} et D_{tot} ne devraient pas nécessairement être connues d'avance par le RM de l'émetteur car la requête de signalisation va les sommer lors de son allée au récepteur.

La négociation préalable entre les applications sur les paramètres de la session à établir (adresse, codec, méthode d'authentification, etc.) ne concerne pas le système.

Une fois reçoit la requête, le système vérifie si le client est authentifié et autorisé à demander ce service (au niveau du plan de service). Si la demande est acceptée, le système la fait passer au RM (au niveau du plan de contrôle). Le RM exécute le contrôle d'admission de bout-en-bout (e2e CAC) qui consiste seulement à constater de l'existence d'un EQ-Path reliant entre les deux domaines source et destination. Si le EQ-Path existe et les informations nécessaires pour la réservation sont disposées le RM lance le processus de réservation déclenchée par l'émetteur, sinon (d_{e2e}, C_{tot} et D_{tot} ne sont pas connus), il lance le processus de réservation déclenchée par le récepteur. Dans tous les cas, la

signalisation est découplée du chemin de données (off-path). La communication entre le RM et l'application de client se fait par l'intermédiaire du module AQ-SSN.

4.3.4 La réservation déclenchée par le récepteur

Le déroulement de réservation déclenchée par le récepteur est comme suit :

- le RM du domaine source fait passer la requête incluant le descripteur de trafic au RM du domaine suivant qui à son tour fait le même jusqu'au RM du domaine destinataire;
- le RM destinataire renvoie au récepteur, à travers le module AQ-SSN, la demande de réservation. En connaissant toutes les informations nécessaires, ce dernier peut facilement calculer la bande passante (R) qui garantit la QoS demandée et renvoie le résultat à son RM;
- le RM effectue le contrôle d'admission de domaine (Domain CAC) afin de vérifier la disponibilité de suffisamment de bande passante et de mémoire pour satisfaire la demande. Si la demande est satisfaisable, le RM mappe les paramètres de réservation (la bande passante R et le buffer b) en paramètres propres à la technologie sous-jacente et les passe au RA. Le RM envoie en même temps le message de signalisation au RM précédent (vers le sens de l'upstream) qui réitère le même travail, et ainsi jusqu'au domaine source;
- le RA réalise la réservation et la configuration requise au niveau des équipements réseau et envoie au RM un message de confirmation de réservation;
- après avoir reçu la confirmation de la part du RA, le RM envoie, lui aussi, un message de confirmation au RM précédent;
- le dernier RM (celui du domaine source) fait le même travail que les autres, mais il n'envoie pas le message de confirmation de réservation à l'émetteur (par l'intermédiation de AQ-SSN) que lorsqu'il reçoit les confirmations de son RA et de son RM successeur à la fois;
- l'émetteur reçoit la confirmation de la réservation et peut alors commencer l'envoie de données.

Le message de rafraîchissement est envoyé périodiquement du récepteur vers son RM qui le renvoie au RM suivant, et ainsi jusqu'au RM destinataire.

A l'intérieur de domaine, le rafraîchissement de réservation dépend du choix de la technologie sous-jacente : Soft Stat ou Hard Stat.

Quand l'émetteur veut terminer la connexion il envoie un message de libération de ressources à son RM. Celui-ci le retransmet au RM suivant et demande du RA de libérer les ressources allouées à la connexion concerné. Lors de l'arrivée du message au récepteur il répond par un message de confirmation qui retourne jusqu'à l'émetteur.

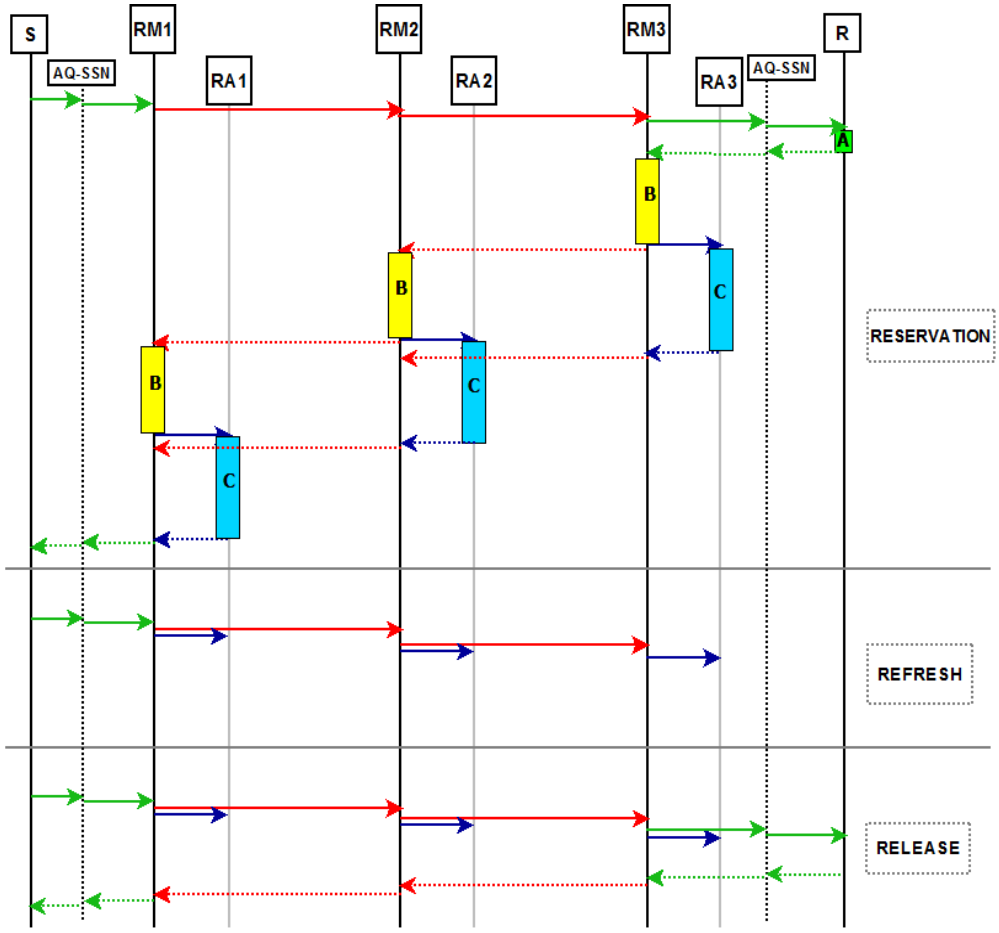


Figure 4.14: réservation déclenchée par le récepteur.

4.3.5 La réservation déclenchée par l'émetteur

La réservation déclenchée par l'émetteur vise à réduire le temps de réservation. Elle suppose que l'émetteur connait d'avance le délai maximum requis pour le trafic qu'il va générer, et le RM source dispose les valeurs des délais dus au traitement au sein des routeurs et à la transmission de paquet sur le canal au long du chemin (D_{tot} et C_{tot}). Dans ce cas le protocole EQ-GBP est modifié pour transporter les deux délais C_{tot} et D_{tot}, et la fonction de composition de paramètres utilisée est SUM qui retourne la somme de délai du chemin parcourues. Les étapes de signalisation est comme suit:

- Le RM de domaine source reçoit la requête contenant le descripteur de trafic et le délai maximum à garantir et, en utilisant les valeurs de C_{tot} et D_{tot}, il déduit la bande passante R et la taille de buffer b à réserver. Ensuite, il effectue le contrôle d'admission de domaine (Domain CAC). Si la requête est admissible, il envoie une requête de réservation au RM suivant contenant, en plus du descripteur de flux, la bande passante R à réserver, et en attend un message de confirmation de réservation. Le RM mappe les paramètres de réservation en paramètres spécifiques à la technologie sous-jacente et
- fait passer la demande au RA.
- Le RA réalise l'allocation et la configuration des équipements réseau et puis retourne un message de confirmation à son RM.
- Lorsque le RM de domaine destinataire reçoit la requête il fait le même travail que les autres RMs mais cette fois il envoie le message de signalisation au récepteur (toujours par l'intermédiation d'AQ-SSN).

- Chaque RM attend deux messages de confirmation : un du RA et l'autre du RM suivant à l'exception du RM destinataire qui attend une confirmation du RA et une autre du récepteur. Lorsque les deux confirmations arrivent, il retourne une confirmation au RM précédent.
- Après avoir reçu les deux messages de confirmation, le RM source retourne une confirmation au module AQ-SSN qui, à son tour, informe l'émetteur que sa demande est acceptée et il peut commencer la communication.

Le processus de rafraîchissement et de libération de ressources reste le même que dans le cas de réservation déclenchée par le récepteur.

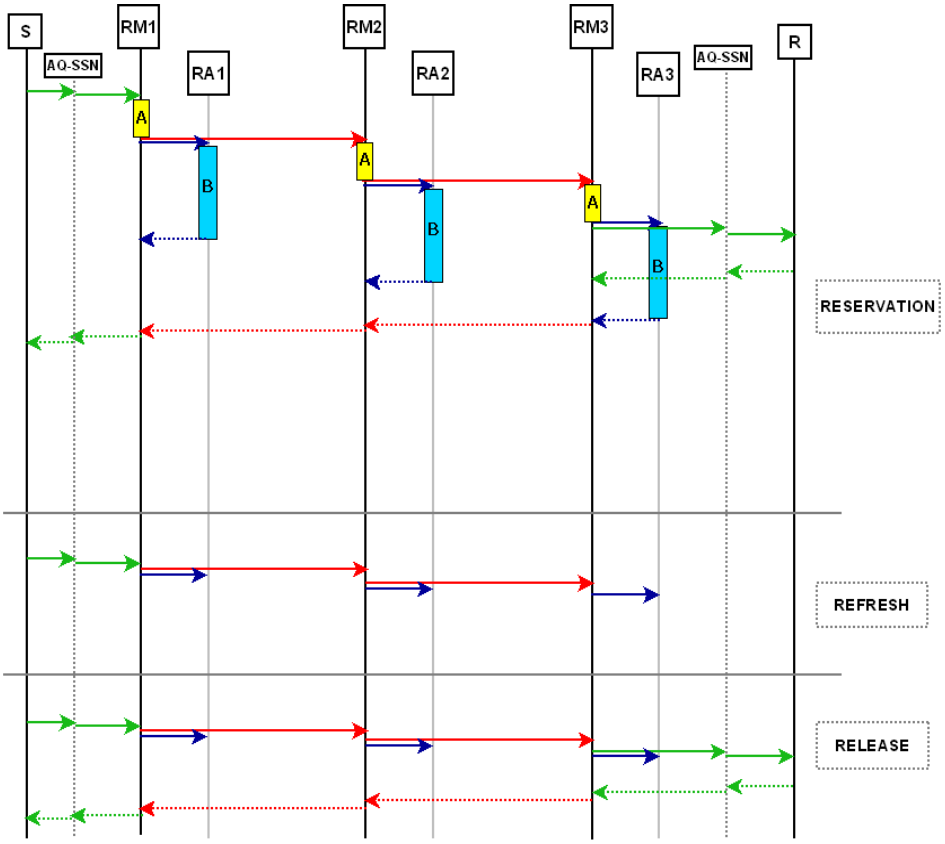


Figure 4.15: réservation déclenchée par l'émetteur.

4.3.6 Conclusion

Le système EuQoS résout le problème de l'hétérogénéité des réseaux en disposant, sur chaque domaine, d'une couche abstraite représentant les ressources et les services offerts par le réseau sous-jacent. Il propose, ainsi, des classes services uniformes entre tous les domaines.

Notre contribution complète cette solution par l'ajout d'un nouveau service qui traite chaque flux individuellement. Ce service par flux permet d'offrir une sorte de ligne virtuelle au QoS de bout-en-bout. Chaque domaine le mappera en service spécifique à sa technologie.

Nous croyons que ce nouveau service sera très intéressant aux entreprises qui cherchent un service Internet satisfaisant à la fois : les paramètres de QoS et les exigences de la sécurité.

Conclusion générale et perspectives

Les travaux sur la qualité de service ont abouti à certaine maturité où ils peuvent assuré la QoS intra-domaine et inter-domaine en proposant des architectures, des mécanismes et des protocoles répondant aux difficultés existantes.

Au niveau intra-domaine, les architectures IntServ, DiffServ, MPLS TE et PBNM sont définies par l'IETF et d'autres organisations de normalisation pour permettre une assurance et une optimisation de la gestion de QoS au sein d'un seul domaine. Elles utilisent une panoplie de mécanismes et de protocoles qui traitent la gestion du trafic et l'optimisation de la gestion des ressources et des services. Au niveau inter-domaine, des extensions ont été proposées pour assurer la QoS de bout-en-bout. Ces extensions couvrent les protocoles d'échange d'informations et de signalisations entre les domaines et les méthodes de coopération entre les domaines dans le but de calculer les chemins de bout-en-bout satisfaisant les besoins en QoS.

Le système EuQoS fait un pas d'avance et propose une solution au problème d'hétérogénéité des réseaux. La solution EuQoS repose sur l'abstraction des services et des processus de signalisation et le mappage dans chaque domaine en services et processus spécifique à la technologie sous-jacente.

La problématique particulière traitée dans ce mémoire est d'assurer la QoS de bout-en-bout dans un environnement hétérogène pour des flux de nature inconnue (CVR, VBR, etc.). La méconnaissance peut être causée par des soucis de sécurités. Nous avons démontré par simulation que l'agrégation d'un flux d'une classe quelconque en une autre classe, semblable en termes de paramètres de QoS assurés, entraîne un effet indésirable sur les flux. Cette conclusion nous a amené à proposer un nouveau service par flux qui gère chacun de ces flux individuellement. De la même philosophie d'EuQoS, le service et le processus de signalisation sont abstraites, et c'est à chaque domaine de les mapper en technologie spécifique sous-jacente.

Comme suite et perspectives à ce que nous avons présenté, démontré et proposé, il sera très intéressant de :

- définir le mappage entre le service par flux et les services de différentes technologie ;
- spécifier le protocole d'échange d'information EQ-BGP modifié ;
- spécifier le processus d'approvisionnement loose ;
- implémenter la solution complète dans le Testbed du système EuQoS et expérimenter son fonctionnement.

Bibliographie

[Bak1] A. Bak, W. Burakowski, M. Fudala, H. Tarasiuk, C. Brandauer, T. Ziegler, M. Markaki, E. Nikolouzou, T. Engel, F. Ricciato et S. Salsano, “Specification of traffic handling for the first rial“, Aquila Project, 21 Juillet 2000.

[Bak2] A. Bak, A. Beben, W. Burakowski, M. Dabrowski, M. Fudala, H. Tarasiuk, et Z. Kopertowski, “AQUILA network architecture: first trial experiments“, Journal of telecommunications and information technology, Février 2002.

[Bertrand] g. Bertrand, S. Lahoud, M. Molnár, G. Texier, "Inter-Domain Path Computation With Multiple Constraints ", Institut De Recherche En Informatique Et Systèmes Aléatoires IRISA, France, Août 2008.

[Braun] T. Braun, M. Diaz, J. E. Gabeiras, T. Staub, "End-to-End Quality of Service Over Heterogeneous Networks", Springer, 2008.

[Bruin] X. Masip-Bruin, M. Yannuzzi, J. Domingo-Pascual, A. Fonte, M. Curado, E. Monteiro, F. Kuipers, P. V. Mieghem, S. Avallone, G. Ventre, P. Aranda-Gutiérrez, M. Hollick, R. Steinmetz, L. Iannone, K. Salamatian, "Research challenges in QoS routing", Computer Communications, Volume 29, Numéro 5, 6 March 2006.

[Chao] H. Jonathan Chao et X. Guo, ”Quality of service control in high-speed networks”, John Wiley & Sons, 2002.

[Curado] M. Curado, E. Monteiro, “A Survey of QoS Routing Algorithms”, International Conference on Information Technology (ICIT2004), Istanbul, Turkey, 17-19 Dec 2004.

[Davie] B. S. Davie, A. Farrel, "MPLS: next steps", Morgan Kaufmann Publishers, USA, 2008.

[Deleuze] C. Deleuze, "Qualité de service dans l'Internet : problèmes liés au haut débit et au facteur d'échelle", thèse PhD, Paris VI, France, 2000.

[E800] ”Recommendation E800. Terms and Definitions Related to QoS and Network Performance Including Dependability”, ITU-T, Août 1994.

[Frikha] A. Frikha, S. Lahoud,"Hybrid Inter-Domain QoS Routing based on Look-Ahead Information", inria-00463460, version 1, INRIA, 11 Mars 2010.

[Garcia] F. Garcia, "Processus Décisionnels de Markov en Intelligence Artificielle", groupe PDMIA, 27 fév. 2008, http://researchers.lille.inria.fr/~munos/papers/files/bouquinPDMIA.pdf.

[Gelenbe] E. Gelenbe, R. Lent et Z. Xu, "Measurement and performance of cognitive packet networks", Jour. Comp. Networks, Vol. 37, 691-701, 2001.

[Guo] L. Guo, I. Matta, “Search Space Reduction in QoS Routing”, 19th IEEE International Conference on Distributed Computing Systems, Austin, Texas, USA, May 31 - June 04, 1999.

[Hoceini] S. Hoceini,"Techniques d’apprentissage par renforcement pour le routage adaptatif dans les réseaux de télécommunication à trafic irrégulier", thèse Doc., L’université Paris XII, France, 23 Nov. 2004.

[Htira] W. Htira, "Découverte et Agrégation de Topologies de Réseaux, application au Contrôle d'admission", thèse doctorat, université de Toulouse, France, 12 Nov. 2008.

[Huston] G. Huston, http://www.cidr-report.org/v6/.

[ISO8402] "Quality Management and Quality Assurance Vocabulary. Technical Report", ISO8402, International Organization for Standardization, 2000.

[Juttner] A. Juttner, B. Szviatovszki, I. Mécs, Z. Rajko, "Lagrange Relaxation Based Method for the QoS Routing Problem", Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2001.

[Kuipers] F. Kuipers, "Quality of Service Routing in the Internet,Theory, Complexity and Algorithms", thèse Phd, université de technologie Delft, Holland, 2004.

[Labovitz] C. Labovitz, G. R. Malan, et F. Jahanian "Internet Routing Instability", Proceedings de ACM SIGCOMM '97 conference on Applications, technologies, architectures, and protocols for computer communicationm, Sep 1997.

[Lochin] E. Lochin, "Implémentation des mécanismes de qualité de service et de contrôle de trafic sous linux", http://manu.lochin.net/qos/qoshtml/qos.html, visité le 22/10/2010.

[Mellouk] A. Mellouk, "End-to-End Quality of service engineering in next generation heterogenous networks", ISTE et John Wiley & Sons, 2009.

[Mieghem] P. V. Mieghem, F. A. Kuipers, "Concepts of Exact QoS Routing Algorithms", IEEE/ACM Transactions On Networking, Vol. 12, Nu. 5, Oct 2004.

[Mingozzi] E. Mingozzi, G. Stea, M. A. Callejo-Rodriguez, J. Enriquez-Gabeiras, G. Garcia-de-Blas, F. J. Ramon-Salquero, W. Burakowski, A. Beben, J. Sliwinski, H. Tarasiuk, O. Dugeon, M. Diaz, L. Baresse, E. Monteiro, "EuQoS: End-to-End Quality of Service over Heterogeneous Networks", Computer Communications, 2009.

[Neve] H. D. Neve, P. V. Mieghem, "TAMCRA: A Tunable Accuracy Multiple Constraints Routing Algorithm", Computer Communications, Vol. 23, 2002.

[PCE-H] D. King, A. Farrel, "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", draft-king-pce-hierarchy-fwk-05, Sept. 2010.

[Peterson] Larry L. Peterson et Bruce S. Davie, "Computer networks: a systems approach", 4ième édition, Morgan Kaufmann Publishers, USA, 2007.

[Pujolle] G. Pujolle, "Les réseaux", 6ième édition, chapitre 19, Eyrolles, 2008.

[QBone01] "Qbone Signaling Design Team, Final Report. Technical Report", QBone Signaling Work Group, 2001.

[Racaru] S. F. Racaru,"Conception et validation d'une architecture de signalization pour la garantie de qualité de service dans l'Internet multi-domaine, multi-technologie et multi-service", thèse doctorat, université de Toulouse, France, 14 Oct 2008.

[RFC1930] J. Hawkinson, T. Bates, "Guidelines for Creation, Selection, and Registration of an Autonomous System (AS) ", RFC 1930, IETF, Mars 1996.

[RFC2327] M. Handley et V. Jacobson, "SDP: Session Description Protocol", RFC 2327, IETF, Avril 1998.

[RFC2330] V. Paxson, G. Almes, J. Mahdavi et M. Mathis, “Framework for IP Performance Metrics“, RFC 2330, IETF, Mai 1998.

[RFC2386] E. Crawley, R. Nair, B. Rajagopalan et H. Sandick, “A Framework for QoS-based Routing in the Internet”, IETF RFC 2386, IETF, Août 1998.

[RFC2597] J. Heinanen, F. Baker, W. Weiss et J. Wroclawski, “Assured Forwarding PHB Group”, RFC 2597, IETF, Juin 1999.

[RFC2598] V. Jacobson, K. Nichols et K. Poduri, “An Expedited Forwarding PHB”, RFC 2598, IETF, Juin 1999.

[RFC2638] N. Nichols, "A Two-bit Differentiated Services Architecture for the Internet", IETF RFC 2638, Juil. 1999.

[RFC3270] F. Le Faucheur, B. Davie, S. Davari, P. Vaanen, R. Krishnan, P. Cheval et J. Heinanen, "Multiprotocol Label Switching (MPLS) Support for Differentiated Services", RFC3270, IETF, Mai 2002.

[RFC3550] H. Schulzrinne, S. Casner, R. Frederick et V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications”, RFC 3550, IETF, Juillet 2003.

[RFC3588] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, J. Arkko, "Diameter Base Protocol", RFC 3588, IETF, Sept. 2003.

[RFC4080] R. Hancock, G. Karagiannis, J. Loughney et S. Van den Bosh, “Next Steps in Signaling (NSIS): Framework”, RFC 4080, IETF, Juin 2005.

[RFC4594] J. Babiarz, K. Chan et F. Baker, "Configuration Guidelines for DiffServ Service Classes", RFC 4594, IETF, Août 2006.

[RFC4655] A. Farrel, J. P. Vasseur, G. Ash. "A Path Computation Element (PCE)- Based Architecture", RFC 4655, IETF, Août 2006.

[RFC5152] J. P. Vasseur, A. Ayyangar, R. Zhang. "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs) ", RFC 5152, IETF, Fév. 2008.

[RFC5440] J. P. Vasseur, J. L. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, IETF, Mars 2009.

[RFC5441] J. P. Vasseur, R. Zhang, N. Bitar, J. L. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", RFC 5441, IETF, Avril 2009.

[Wang] Z. Wang, "Internet QoS: architectures and mechanisms for Quality of Service", Morgan Kaufmann Publishers, USA, 2001.

[X902] "Information Technology-Open Distributed Processing-Reference Model: Foundations, technical Report", X.902, International Telecommunication Union, 1995.

[XiPeng] X. XiPeng, “Technical, commercial, and regulatory challenges of QoS”, Morgan Kaufmann Publishers, USA, 2008.

[Yuan] X. Yuan, W. Zheng, S. Ding, “A Comparative Study of QoS Routing Schemes That Tolerate Imprecise State Information”, Eleventh International Conference on Computer Communications and Networks, 2002.

[Zhang] D. Zhang, D. Ionescu, "QoS Performance Analysis in Deployment of DiffServ-aware MPLS Traffic Engineering", Proceedings of the Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, v. 03, p. 963-967, Août 2007.