Uma Arquitetura de Roteamento com Qualidade de Serviço Baseada em Redes Ativas[1]

Flavio Paiva Junqueira
flavio@gta.ufrj.br

Luís Henrique M. Kosmalski Costa
luish@gta.ufrj.br

Otto Carlos Muniz Bandeira Duarte
otto@gta.ufrj.br

Grupo de Teleinformática e Automação - GTA
Universidade Federal do Rio de Janeiro
COPPE/EE - Programa de Engenharia Elétrica/Departamento de Eletrônica
Caixa Postal 68504 - CEP 21945-970 - Rio de Janeiro - RJ - Brasil

Resumo

Neste artigo, é proposta uma arquitetura de referência para roteadores onde é utilizado o paradigma de redes ativas para a implementação dos mecanismos de troca de informação necessários para o roteamento baseado em QoS. Fragmentos de programa, contidos em cápsulas, percorrem os nós da rede com três objetivos: descobrir a topologia da rede, atualizar o estado dos enlaces e alterar o comportamento dos roteadores ativos. Esta arquitetura está baseada no modelo de Serviços Integrados para a Internet, especificado pelo IETF.

Abstract

This paper presents a reference architecture which uses the active networking paradigm to implement the mechanisms needed by QoS-based routing. Program fragments move across the network nodes to achieve three goals: discover the network topology, update link states, and change the active router behavior. This architecture is based on the Integrated Services model in the Internet defined by the IETF.

[1] Este trabalho foi realizado com recursos da UFRJ, FUJB, PROTEM-CC, CNPq, FAPERJ, CAPES e COFECUB.

1 Introdução

Com o surgimento de novas aplicações multimídias na Internet, a disciplina de melhor esforço utilizada pelo protocolo IP (*Internet Protocol*) torna-se inadequada, dados os requisitos de Qualidade de Serviço (QoS) destas aplicações. Algumas propostas vêm sendo feitas no que diz respeito à definição de uma arquitetura de Serviços Integrados para a Internet [1]. Dentre estas, destacam-se a implementação do protocolo IP sobre a rede ATM (*Asynchronous Transfer Mode*), e a sua integração com o protocolo proposto pelo IETF (*Internet Engineering Task Force*) para a realização de reservas de recursos, o RSVP (*Resource ReSerVation Protocol*) [2]. Para considerar novas classes de serviço, o protocolo de roteamento deve ser modificado. A utilização do paradigma de redes programáveis, isto é, a solução por redes ativas, mostra-se uma alternativa na implementação destes novos protocolos de roteamento [3].

O roteamento baseado em requisitos de QoS envolve a modificação dinâmica das métricas associadas às rotas, de acordo com as mudanças ocorridas nos parâmetros de QoS dos enlaces [4, 5, 6]. Com isto torna-se necessária a atualização periódica das tabelas de rotas. A solução proposta neste trabalho utiliza códigos executáveis que percorrem os nós da rede atualizando as tabelas de roteamento.

O percurso de códigos executáveis pelos nós da rede condiz com os conceitos apresentados nas propostas de redes ativas. As redes ativas permitem a injeção de programas adaptados às necessidades da aplicação nos nós da rede, possibilitando a implantação de novos serviços de forma rápida, flexível e de baixo custo [3, 7, 8, 9, 10]. A proposta realizada neste trabalho utiliza o modelo integrado [9, 10] para programação dos nós da rede, onde pacotes convencionais são substituídos por cápsulas, que contêm código executável e dados. A execução destas cápsulas deve ser possível em qualquer nó ativo da rede, obedecendo a um modelo computacional comum.

A arquitetura de implementação proposta neste trabalho baseia-se nos conceitos de redes ativas e nos requisitos do roteamento baseado em QoS. Em cada nó ativo existe uma máquina de processamento [10], que opera em conjunto com a camada IP e o protocolo de roteamento. Neste contexto, o mecanismo de cápsulas é utilizado com dois objetivos principais: a atualização das tabelas de roteamento e a descoberta de rotas no estabelecimento de conexão. A cada alteração na qualidade de serviço dos enlaces conectados a um nó, são enviadas cápsulas que percorrem a rede alterando as tabelas de roteamento dos outros nós. Este envio assíncrono permite que a quantidade de informação gerada se adeqüe à dinâmica da rede.

Este artigo está estruturado da seguinte forma. Na seção 2, os principais requisitos do roteamento baseado em QoS são apresentados. Os diversos aspectos do modelo computacional de redes ativas são abordados na seção 3. A proposta de arquitetura de referência para roteamento ativo é apresentada na seção 4. Por último, na seção 5, são feitas algumas considerações finais.

2 Roteamento Baseado em Requisitos de QoS

Considerando a organização da Internet, a natureza do tráfego gerado, as classes de serviço que devem ser providas e os requisitos dos usuários, podem ser definidos diversos pré-requisitos ao roteamento baseado em reservas de recursos. A seguir são apresentados diversos requisitos, divididos nos grupos principais: métricas e rotas, robustez, eficiência, escalabilidade, prioridades, inter-domínios e políticas de controle de tráfego.

Quanto a métricas e rotas, é desejável que o protocolo de roteamento suporte múltiplas métricas para os caminhos (atraso, faixa de transmissão, variação de atraso etc.). A definição das métricas necessárias deve ser realizada com base nas classes de serviço suportadas [4, 5]. O protocolo deve ser capaz de calcular dinamicamente rotas satisfazendo múltiplas restrições. Deve oferecer suporte de QoS para vários tipos de fluxos, incluindo ponto-a-ponto (*unicast*) e ponto-a-multiponto (*multicast*) e, além disso, deve suportar receptores heterogêneos em termos dos requisitos de QoS. Além disso, protocolo deve ser robusto, ou seja, possuir as habilidades de detectar e responder a mudanças nas capacidades de QoS dos enlaces e de continuar o roteamento mesmo diante de falhas múltiplas na rede.

A utilização dos recursos da rede deve ser eficiente, através de balanceamento da carga e controle de admissão global de chamadas. O controle de admissão local de chamadas é necessário para administrar os recursos em cada roteador, não levando em conta qualquer visão global do consumo de recursos pelos diversos fluxos. Pode-se obter uma melhor eficiência no roteamento acrescentando-se um módulo de controle de admissão. Este controle de admissão de mais alto nível pode considerar o consumo de recursos de cada fluxo em relação aos recursos disponíveis ao longo de um caminho de forma a determinar se a aceitação deste fluxo é desejável globalmente. O Grupo de Trabalho em Serviços Integrados do IETF vem se concentrando no gerenciamento local de recursos nos roteadores.

A sobrecarga de processamento devida ao roteamento deve ser mínima. Mesmo sem o roteamento baseado em QoS, as sobrecargas devidas ao estabelecimento de fluxo através do RSVP são adicionadas em cada sessão. A

diminuição desta sobrecarga deve ser buscada. Uma das formas possíveis pode ser a utilização de *caches* com as rotas recentemente computadas, de maneira que novos fluxos para os mesmos destinos não exijam novo cálculo dos caminhos. Deve ser possível o controle dos recursos, ou seja, a capacidade de limitar o consumo de recursos da rede pelas diferentes classes de tráfego.

O protocolo deve ser escalável, pré-requisito indispensável quando se considera a dimensão de uma rede como a Internet. O algoritmo utilizado deve ser capaz de lidar com o aumento do número de nós e de enlaces da rede sem se tornar inviável em termos da capacidade de processamento das máquinas [5 ,6].

Deve existir um mecanismo para estabelecer prioridades de transmissão entre os fluxos, determinar se um fluxo tem o direito de utilizar a prioridade requisitada e permitir que tráfegos de alta prioridade sejam estabelecidos com precedência em relação aos menos prioritários. Assume-se que o RSVP possa ser utilizado para sinalizar a prioridade de um fluxo [4].

Os requisitos para o roteamento entre domínios diferentes constituem um aspecto importante que tem gerado bastante discussão. Mais especificamente, a natureza da informação trocada na fronteira entre dois domínios administrativos deve ser determinada.

O cálculo de rotas para a acomodação de fluxos ponto-a-multiponto com requisitos de QoS é um problema delicado, especialmente com a noção de um controle de admissão global. O dinamismo permitido pelo IP *Multicast* no grupo de receptores, aliado à heterogeneidade dos receptores tornam o problema mais complexo. Considerando um esquema do tipo adotado pelo MOSPF (*Multicast Open Shortest Path First*) [11] a principal dificuldade diz respeito à escalabilidade. Para considerar QoS, o cálculo do caminho *multicast* no roteador deve levar em conta conhecimento não apenas da identidade das sub-redes onde membros do grupo estão presentes, mas também a identificação dos ramos na árvore existente. Ou seja, nesta versão do MOSPF, os roteadores precisam manter informação específica das fontes para cada grupo. Mudanças no grupo de destinos ou reservas realizadas por receptores resultam em informação sobre múltiplas árvores sendo distribuídas para todos os roteadores na rede.

2.1 Propagação da informação

A arquitetura de roteamento baseada no estado do enlace requer que os roteadores informem através de difusão o *status* dos recursos locais (como a faixa de transmissão disponível e os atrasos medidos) e as informações relacionadas com a topologia local para todos os outros roteadores da rede. A difusão da informação de *status* de um roteador para os outros tende a ser um processo dispendioso em termos de sobrecarga do sistema de comunicação e das

capacidades de processamento. Até então, o roteamento intra-domínio tem sido baseado em métricas fixas para os enlaces. Neste contexto, a eficiência na propagação da informação não era um pré-requisito. Por esse motivo, os algoritmos de roteamento como o implementado pelo OSPF utilizam-se de inundação com um período de *refresh* como forma de propagar as atualizações.

Para o roteamento baseado em QoS eficiente, no entanto, mecanismos de inundação não são adequados para a distribuição da informação. Técnicas alternativas para reduzir a sobrecarga da difusão, como o envio baseado em árvores, foram propostas. Além disso, a informação do *status* dos enlaces é em muitos casos volátil, ou seja, seus valores podem mudar significativamente em curtos intervalos de tempo com a admissão de fluxos de tempo real. A maneira como essa informação é quantificada de forma a diminuir a freqüência das atualizações é uma questão importante. Quanto menos precisa for a informação, menor a chance de um roteador fonte encontrar um caminho ideal. Por outro lado, os custos da manutenção de informação muito acurada em cada roteador podem ser muito altos.

Uma forma alternativa de regular a distribuição da informação, apresentada neste trabalho, consiste em utilizar o modelo de redes ativas na sua propagação, através de fragmentos de programa que percorrem os nós da rede realizando as atualizações necessárias ao protocolo de roteamento.

3 Aspectos de Redes Ativas

Uma visão extrema de redes ativas é aquela na qual toda mensagem é um programa. Toda mensagem, neste caso denominada cápsula, que passa nos nós contém um fragmento de programa (de pelo menos uma instrução) que pode incluir dados embutidos. Quando uma cápsula chega a um nó ativo, seu conteúdo é avaliado da mesma forma que um dispositivo Postscript interpreta o conteúdo de cada arquivo que lhe é enviado [3, 9].

A figura 1 fornece uma visão conceitual de como um nó ativo deve ser organizado. Ao ser recebido por um nó ativo, o conteúdo da cápsula é enviado para um **ambiente de execução transiente** onde será avaliado de forma segura. Supõe-se que os programas são compostos de instruções primitivas, que realizam computações básicas no conteúdo da cápsula, e podem ainda chamar métodos externos, os quais fornecem acesso controlado aos recursos fora do ambiente transiente. A execução de uma cápsula pode resultar no escalonamento de zero ou mais cápsulas para transmissão nos enlaces de saída e pode alterar o estado não-transiente do nó. O ambiente transiente é finalizado quando a execução da cápsula termina, sendo reutilizado para a execução de

outras cápsulas.

No modelo integrado, o processamento realizado nos nós da rede sobre as cápsulas é baseado nos programas que estas cápsulas contém. A execução destes programas só é possível se houver um acordo com relação ao modelo computacional adotado por todos os nós ativos da rede. Neste contexto, o modelo computacional é definido como sendo o conjunto de instruções e recursos disponíveis para os programas contidos nas cápsulas [3].

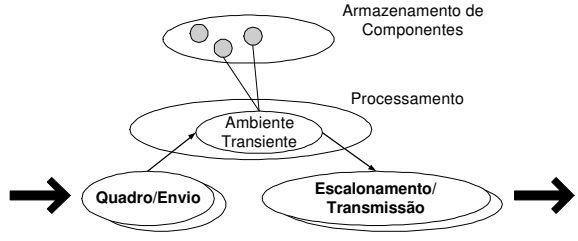

Figura 1 – Esquema de execução em um nó ativo

Dentre os recursos disponíveis nos nós, nem todos estarão disponíveis para todos os programas encapsulados. O acesso a determinados recursos dependerá de alguma forma de autenticação, e estará disponível somente para um conjunto de sistemas fonte. O conjunto de instruções é baseado no modelo de programação adotado.

A codificação dos programas encapsulados pode ser realizada através de código fonte, código intermediário ou código de máquina. A utilização de código fonte implica na interpretação do programa, isto é, uma conversão é realizada em tempo de execução da linguagem de alto nível para o código da máquina local. As principais vantagens deste tipo de codificação são a facilidade de implementação e a portabilidade. Por outro lado, esta conversão realizada em tempo de execução acarreta uma degradação no desempenho. Códigos intermediários como Java são interpretados também, mas possuem a vantagem de terem, realizadas antes do tempo de execução, as fases de pré-processamento, análise léxica e checagem estática [12]. Com isto, mantém-se a portabilidade aumentando-se o desempenho. O último tipo de codificação é por código de máquina, onde os programas são escritos em uma linguagem de alto nível e compilados para uma arquitetura particular. Apesar do ganho em desempenho na execução, perde-se portabilidade.

Dentro do ambiente de execução, em uma abordagem restrita, as cápsulas não teriam acesso a recursos além do ambiente transiente de execução. Embora

isto seja suficiente para o propósito de diversas aplicações, algumas aplicações requerem o acesso a recursos além do ambiente transiente. Programas podem acessar tais recursos através de:

- **componentes de fundação**: são métodos externos que implementam o acesso controlado dos programas aos recursos disponíveis nos nós;
- **armazenamento ativo**: é um espaço de memória não-transiente onde uma cápsula pertencente a um fluxo pode consultar e atualizar o estado de um fluxo. Este estado permanece armazenado temporariamente se não houver atualização, e é conhecido como *soft state*;
- **extensões**: cápsulas podem carregar e disponibilizar métodos e classes para as execuções posteriores.

A interoperabilidade entre os diversos nós ativos, no que diz respeito aos programas encapsulados, é conseguida através do estabelecimento de um conjunto de abstrações dos recursos disponíveis. Todos os nós ativos devem respeitar a nomeação dada para estas abstrações, sendo as questões de eficiência e segurança problemas de implementação local.

A partir da identificação dos recursos disponibilizados, pode-se particionar as atividades de gerenciamento dos recursos em três tipos:

- **atribuição dinâmica**: os recursos são acessados por métodos externos e, portanto, o conjunto de recursos disponível é dado pelo confinamento no qual a cápsula foi inicialmente colocada.;
- **validação**: o código encapsulado deve ser validado para que o nó conceda o acesso aos recursos;
- **delegação**: existe a necessidade de um mecanismo de delegação automática de autorizações de acessos a recursos para a expansão do conjunto de inicialmente disponível através de autenticação.

4 Arquitetura de Roteamento Ativo – GTA/RA

A arquitetura de roteamento ativo proposta neste artigo baseia-se nas propostas para IP ativo e para Serviços Integrados na Internet. A arquitetura de referência para implementação da máquina de processamento utilizada com a opção ativa do IP é definida nesta seção.

4.1 IP Ativo

Tennenhouse *et al.* propõem uma extensão do mecanismo de opções do protocolo de rede IP de forma que seja possível o transporte de fragmentos de programas [10]. O mecanismo de opções do IP foi utilizado como ponto de

partida, pois existe justamente para acomodar um processamento agregado que não está relacionado com o caso comum. As opções do IP têm sido utilizadas para processamento não usual de datagramas, tipicamente para monitorar e medir a rede. O modelo de processamento para a opção ativa é mostrado na figura 2. A máquina de processamento (MP) é adjacente à camada IP, e é invocada de dentro desta para a execução dos programas encapsulados.

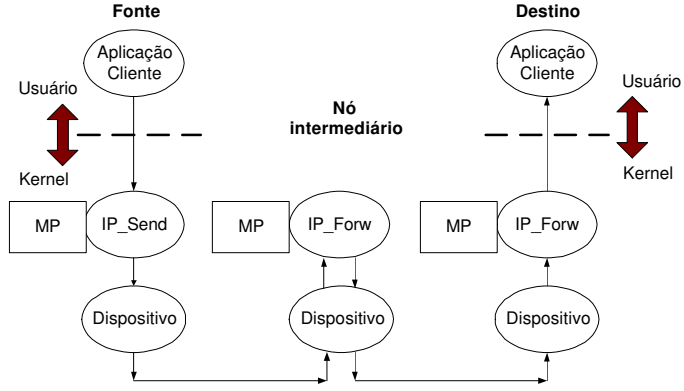

Figura 2 – Processamento em Nós Ativos

A arquitetura de referência para a máquina de processamento é proposta em [13]. Esta arquitetura utiliza a linguagem Java para programação das cápsulas, e constitui-se principalmente de um **Núcleo de Máquinas Virtuais** (NMV) e de módulos auxiliares que, junto com o núcleo, cumprem os aspectos apresentados na seção 3 para a obtenção de um modelo computacional seguro, robusto e eficiente. O diagrama da arquitetura de implementação proposta é apresentado na figura 3.

O Núcleo de Máquinas Virtuais gerencia a abertura de máquinas virtuais baseadas em Java para as cápsulas que lhe são entregues. Cada cápsula tem uma máquina virtual aberta para a sua execução. Os recursos disponibilizados para a cápsula na criação da máquina virtual vão depender da identificação da origem da cápsula. Assume-se que a identificação é feita utilizando-se algum mecanismo de criptografia [3]. A autenticação é realizada por um **Gerenciador de Autenticações** (GA), que mantém informações sobre os privilégios que os sistemas fonte têm em termos de execução.

Em conjunto com o GA, o **Gerenciador de Recursos** (GR) provê o acesso propriamente dito. Todo pedido de acesso a um recurso especial do nó como

acesso a tabelas de roteamento ou armazenamento não-transiente deve ter os privilégios conferidos junto ao GA para ser realizado.

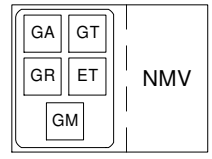

Figura 3 – Máquina de Processamento

Os três últimos módulos foram baseados na arquitetura de implementação para sistemas de comunicação de alto desempenho GTA/MM [14]. Esta arquitetura citada possui três módulos básicos que são: o **Gerenciador de Memória** (GM), o **Gerenciador de Temporizações** (GT) e o **Escalonador de Tarefas** (ET). O GM é baseado em um mecanismo de *garbage collection*. O acesso é promovido segundo as operações básicas definidas para este mecanismo. O GT fornece temporizadores para ações como invalidar cápsulas que foram postas para "dormir" e assim permaneceram indefinidamente. Por último, o ET realiza a alternância entre os processos em execução na máquina de processamento.

4.2 A Arquitetura GTA/RA

O modelo proposto neste artigo baseia-se na extensão da arquitetura de Serviços Integrados na Internet, definida pelo IETF [1]. Esta extensão propõe-se a suportar aplicações de tempo real, controle de compartilhamento de enlaces, além do serviço de melhor esforço. Para isto, a arquitetura de Serviços Integrados define uma função no roteador responsável por fornecer diferentes requisitos de qualidade de serviço para os diversos fluxos, denominada **controle de tráfego**. A funcionalidade deste controle é definida por seis componentes:

- **controle de admissão**: implementa o algoritmo de decisão utilizado para determinar se a QoS requisitada por um novo fluxo pode ser provida;
- **controle de políticas**: responsável por tomar decisões administrativas com relação à aceitação de fluxos;
- **classificador**: para propósitos de controle de tráfego, cada pacote deve ser mapeado em alguma classe. Todos os pacotes na mesma classe recebem o mesmo tratamento do escalonador de pacotes. O classificador existe dentro do módulo de transmissão do IP (IP_Forwarder);
- **escalonador de pacotes**: gerencia a transmissão de pacotes de diferentes

fluxos utilizando um conjunto de filas, representativas das classes. Para cada interface de saída, existe um conjunto destas filas;

- **reserva de recursos**: processo necessário para criar e manter o estado específico de cada fluxo nas estações e nos roteadores ao longo do caminho. Este componente consiste principalmente do protocolo de reserva de recursos RSVP [2];
- **roteamento baseado em requisitos de QoS**: processo que interage com o RSVP para determinar os caminhos dos fluxos.

A arquitetura GTA/RA consiste dos elementos presentes no modelo de Serviços Integrados para a Internet e inclui módulos para realização e manutenção do roteamento ativo. Seu esquema funcional está representado na figura 4.

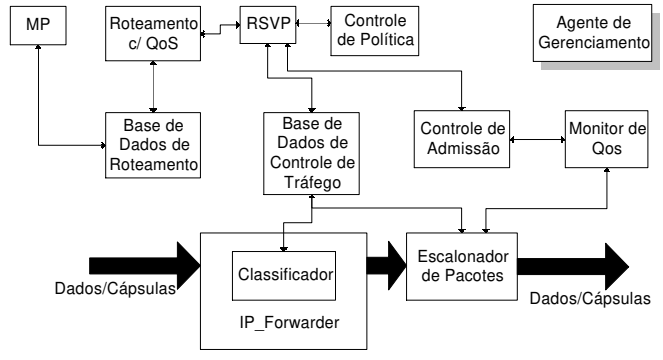

Figura 4 – Arquitetura de Referência para Roteamento Ativo

Na arquitetura proposta, são inseridos três elementos: agente de gerenciamento, máquina de processamento e monitor de QoS. Para o agente de gerenciamento, propõe-se a sua utilização com cápsulas para atualizar as bases de dados dos mecanismos de controle de políticas e de roteamento. A máquina de processamento utilizada baseia-se na especificada na seção 4.1. Toda cápsula recebida pelo nó é executada na máquina de processamento, após passar por um processo de autenticação, e em seguida uma ou mais cápsulas geradas são transmitidas. Neste modelo são definidos três tipos de cápsulas: de **gerenciamento**, de **descoberta de topologia** e de **atualização de topologia**. A execução de cápsulas de gerenciamento causa a modificação do comportamento do roteador através do agente de gerenciamento. Estas cápsulas são geradas por uma aplicação de gerenciamento remota. Os outros dois tipos são utilizados pelo protocolo de roteamento. No modelo é assumido um protocolo de roteamento

que tem conhecimento da topologia da rede e calcula as rotas a partir desta topologia (ex. OSPF). As cápsulas para descoberta de topologia são geradas pelo processo de roteamento, no momento em que um roteador é iniciado. As cápsulas de atualização são geradas pelo monitor de QoS no momento em que alterações significativas são percebidas nas características de QoS dos enlaces de saída de um nó. Este monitor é responsável ainda por fornecer dados para o controle de admissão.

A utilização de cápsulas na atualização assíncrona das tabelas de roteamento permite que pouca informação de controle seja gerada quando a dinâmica da rede estiver mais estável. Por outro lado, quando os estados dos enlaces estiverem variando muito, o volume de dados de controle gerado não precisa crescer na mesma proporção, podendo-se diminuir a acurácia da informação para evitar sobrecarga da rede.

O modelo proposto é basicamente funcional, sendo a implementação responsável por sua otimização. Alguns módulos podem ser implementados dentro de um mesmo processo, como por exemplo os controles de admissão e de políticas, assim como suas bases de dados podem ser unificadas.

5 Conclusões

As novas aplicações multimídias exigem a definição de novos modelos de serviço além do serviço de melhor esforço oferecido atualmente pelo IP. Um componente fundamental para a implementação deste esquema é uma arquitetura de roteamento baseada nos requisitos de QoS dos fluxos transportados.

Neste artigo foi proposta a utilização dos conceitos de redes ativas para a implementação dos mecanismos que constituem o processo de roteamento, mais especificamente através da utilização de cápsulas. Na arquitetura definida para roteamento ativo foram definidas cápsulas para descobrimento e atualização da base de dados de roteamento e para atividades de gerenciamento. A definição da arquitetura GTA/RA foi feita em concordância com o modelo de Serviços Integrados definido pelo IETF.

Para trabalhos futuros, propõe-se que um protótipo desta arquitetura seja implementada, onde devem ser definidas as primitivas para programação das cápsulas junto com os algoritmos necessários para a implementação dos códigos presentes nas cápsulas. A máquina virtual Java utilizada pode ser simplificada para diminuir a sua sobrecarga de processamento.

Referências

[1] R. Braden, D. Clark, S. Shenker, RFC1633: Integrated Services in the Internet Architecture: an Overview. *N. W. G. - IETF*, julho de 1994.

[2] L. Zhang, S. Deering, D. Estrin, S. Shenker e D. Zappala, "RSVP: A New Resource ReSerVation Protocol", *IEEE Network Magazine*, setembro de 1993.

[3] D. Tennenhouse e D. Wetherall, "Towards an Active Network Architecture", *Computer Communication Review*, abril de 1996.

[4] L. H. M. K. Costa e O. C. M. B. Duarte, Roteamento Baseado em Requisitos de Qualidade de Serviço, Relatório Técnico, PEE-COPPE/UFRJ, 1997.

[5] Z. Wang e J. Crowcroft, "Quality of Service Routing for Supporting Multimedia Applications", *IEEE Journal on Selected Areas in Communications*, vol. 14, nº. 7, pp. 1228-1234, setembro de 1996.

[6] Q. Ma e P. Steenkiste, "Quality of Service Routing for Traffic with Performance Guarantees", *IFIP Fifth International Workshop on Quality of Service*, New York, pp. 115-126, maio de 1997.

[7] S. Bhattacharjee, K. Calvert e E. W. Zegura, "An Architecture for Active Networking", *High Performance Network*, White Plains, N. Y., abril de 1997.

[8] T. Yemini e S. da Silva, "Towards Programmable Networks", *IFIP/IEEE International Workshop on Distributed Systems*, outubro de 1996.

[9] D. Tennenhouse *et al*., "A Survey of Active Network Research", *IEEE Communications Magazine*, pp. 80-86, janeiro de 1997.

[10] D. Wetherall e D. Tennenhouse, "The Active IP Option", *Proceedings of the 7th SIGOPS Workshop*, Connemara, Irlanda, setembro de 1996.

[11] J. Moy, RFC1584: Multicast Extensions to OSPF. *N.W.G. - IETF*, março de 1994.

[12] Sun Microsystems Inc., *The Java Virtual Machine Specification*, 1995.

[13] F. Junqueira, O. C. M. B. Duarte, "Uma Arquitetura de Implementação para Redes Ativas", Relatório Técnico, PEE-COPPE/UFRJ, 1997.

[14] C. Albuquerque, "Arquitetura de Implementação de Alto Desempenho para Sistemas Multimídias", Tese de Mestrado, COPPE/UFRJ, abril de 1995.