Preenchendo o Vazio entre Comunicação em Grupo e Multicast Escalável

Maglan Cristiano Diemer Marinho Pilla Barcellos

PIPCA- Programa de Pós-Graduação em Computação Aplicada
C6 - Centro de Ciências Exatas e Tecnológicas
UNISINOS - Universidade do Vale do Rio dos Sinos
Av. Unisinos, 950 - São Leopoldo, RS - CEP 93.022-000 - BRASIL
{maglan,marinho}@exatas.unisinos.br

Resumo

Certos sistemas de comunicação em grupo necessitam de um protocolo de transporte multicast subjacente para operarem de forma eficiente sobre a Internet. Por outro lado, protocolos multicast não oferecem a confiablidade e as garantias desejadas para aplicações distribuídas. Em geral, os sistemas de comunicação em grupo oferecem confiabilidade mas não são escaláveis. Este trabalho tem o objetivo de aproximar os protocolos multicast escaláveis dos sistemas de comunicação em grupo propondo uma arquitetura para trabalharem juntos. Será utilizado o sistema de comunicação em grupo NewTOP ([11]) e o protocolo multicast PRMP ([4]).
Palavras-chaves: comunicação em grupo, protocolo multicast, tolerância a falhas.

1 Introdução

Desde a década de 80 diversos grupos de pesquisa estudam protocolos e *sistemas de comunicação em grupo* e seu emprego em aplicações distribuídas tolerantes a falhas. Muitas contribuições resultaram (e continuam a resultar) de tal esforço, como por exemplo [1], [23], [15], [13] e [2], apenas para citar alguns. Os sistemas de comunicação em grupo fornecem o suporte e a confiabilidade necessários para que aplicações troquem informações de maneira consistente entre membros de grupos, apesar da ocorrência de falhas. Para que esses protocolos e sistemas operem de maneira eficiente na Internet, muitos são os desafios a serem vencidos. Exemplos de esforços nesse sentido são [6], [22] e [14].

Dada a tradicional filosofia de projeto de separar a complexidade em camadas, é desejável a utilização de um *protocolo de transporte multicast* subjacente, que ofereça serviços básicos como controle de erro (detecção de perdas e retransmissão) e congestionamento (convivência com outros fluxos na Internet), entre outros ([3]). Mas estes protocolos de transporte multicast enfrentam problemas de escalabilidade, em particular se confirmações de recebimento (ACKs) são necessárias (múltiplos fluxos TCP, a maneira mais simples e popular, é pior ainda). Antes que comunicação em grupo com tolerância a falhas possa ser empregada com sucesso em larga escala na Internet, sistemas de comunicação em

grupo devem ser combinados com protocolos de transporte multicast e cuidadosamente avaliados em configurações representativas da Internet.

Neste trabalho, descreve-se uma iniciativa cujo objetivo é combinar em uma arquitetura um protocolo escalável para transporte multicast (PRMP [3, 4]) com um sistema de comunicação em grupo que oferece ordenamento total e um sistema de controle de composição de grupo (NEWTOP [16, 18, 17, 11]). Com essa arquitetura, pretende-se implementar uma versão do NEWTOP com PRMP, baseando-se na versão atual do protótipo do NEWTOP ([16]). Tal implementação será utilizada para executar simulações em configurações de rede de larga escala (em número de nós e latências), avaliando-se seu comportamento e escalabilidade. Adicionalmente, a arquitetura será avaliada em cenários com falha de colapso, valendo-se de uma extensão de um conhecido simulador de redes.

O restante deste artigo está organizado da seguinte maneira. A Seção 2 aprofunda e diferencia os conceitos de *comunicação em grupo* e *transporte multicast escalável*. A Seção 3 descreve os princípios básicos e a implementação do NEWTOP, enquanto a Seção 4 discute os controles e o funcionamento do protocolo PRMP. A Seção 5 define uma proposta de arquitetura para a utilização do PRMP como suporte, na camada de transporte, para o sistema de comunicação em grupo NEWTOP. Por fim, a Seção 6 fecha o artigo com as considerações finais e trabalhos futuros.

2 Comunicação em Grupo x Transporte Multicast Escalável

Como acima citado, os sistemas de comunicação em grupo necessitam de uma camada de transporte multicast com garantias mínimas de confiabilidade, para que possam operar eficientemente em aplicações na escala da Internet. O termo *sistema de comunicação em grupo* é usado para representar os protocolos multicast *confiáveis* que gerenciam a comunicação entre os seus membros, possibilitando a troca de mensagens segundo o modelo vários-para-vários. O termo *protocolo de transporte multicast escalável* corresponde ao protocolo de transporte que oferece garantias mínimas de confiabilidade, como por exemplo confirmação positiva de recebimento (ACK). O protocolo de transporte será responsável pelo envio eficiente e escalável de mensagens segundo o modelo um-para-vários, operando sobre a arquitetura de IP multicast. Baixando ainda mais um nível (rede), IP multicast é formado por duas partes bem distintas: protocolos de roteamento multicast inter-redes (como MOSPF ou PIM-SM), e o protocolo de "gerência de grupo" intra-redes (IGMP). Ambos componentes não oferecem qualquer garantia de confiabilidade: operações podem falhar silenciosamente ou atrasar por períodos arbitrários. Os próximos parágrafos esclarecem os serviços prestados pelos *sistemas de comunicação em grupo* e dos *protocolos de transporte multicast*, respectivamente.

Em geral, *sistemas de comunicação em grupo* podem oferecer garantias de entrega de mensagens tais como atomicidade e ordenamento, além de controle consistente de composição de grupo. A atomicidade garante que uma mensagem, uma vez entregue a um processo (*membro*) de um determinado grupo, deve também ser entregue a todos os outros processos em funcionamento do mesmo grupo, mesmo que o processo que originou a mensagem falhe antes de finalizar a transmissão ([8, 19]). O ordenamento das mensagens é necessário para se obter o comportamento correto dos membros dos grupos, sincronizando

a ordem em que as ações são executadas. Há dois tipos comuns de ordenamento utilizados pelos sistemas de comunicação em grupo: ordenamento causal e ordenamento total ([12, 9]). O controle de grupo corresponde ao gerenciamento dos membros de um grupo: criação e destruição de grupos, inserção e remoção de membros. O controle de grupo também é responsável por manter atualizada a informação referente aos membros que compõem um grupo ([7, 10, 17]), atualizando-a consistentemente em caso de falhas. Estas facilidades são importantes para auxiliar a construção de sistemas distribuídos complexos, em particular com requisitos de *dependabilidade.* Entretanto, protocolos e algoritmos para ordenamento e controle de membros existentes nos *sistemas de comunicação em grupo* foram projetados para redes locais, e não são escaláveis ([24]).

Por outro lado, os *protocolos de transporte multicast* não possuem os serviços básicos de comunicação em grupo ([3]). Os *protocolos de transporte multicast* são baseados em um serviço UDP de "melhor esforço", ou seja, tentam entregar a mensagem ao seu destino sem oferecer absolutamente nenhuma garantia. Existem muitos exemplos de protocolos de transporte multicast, estando os mesmos divididos nas classes *orientado-a-remetente* e *orientado-a-receptor* ([25]). Protocolos orientado-a-remetente são baseados em ACKs (confirmações positivas), o qual leva ao problema da implosão (*feedback implosion*). Protocolos orientado-a-receptor, em contraste, visam escalabilidade em detrimento da confiabilidade ou desempenho. Baseados em NACKs (confirmações negativas), estes protocolos passam a **responsabilidade de detectar e recuperar perdas para o receptor**, de forma que o remetente **não mantenha quaisquer informações sobre os receptores**. Por essa razão, esta classe de protocolos não oferece garantias de confiabilidade "fim-a-fim" desejáveis a uma camada subjacente de suporte à comunicação em grupo.

O *protocolo de transporte multicast* baseado em ACKs pode oferecer as garantias desejadas pelo *sistema de comunicação em grupo*, mas, como já mencionado, possui o problema da implosão. Uma alternativa intermediária são os protocolos baseados em *polling* (veja Seção 4).

3 NewTOP

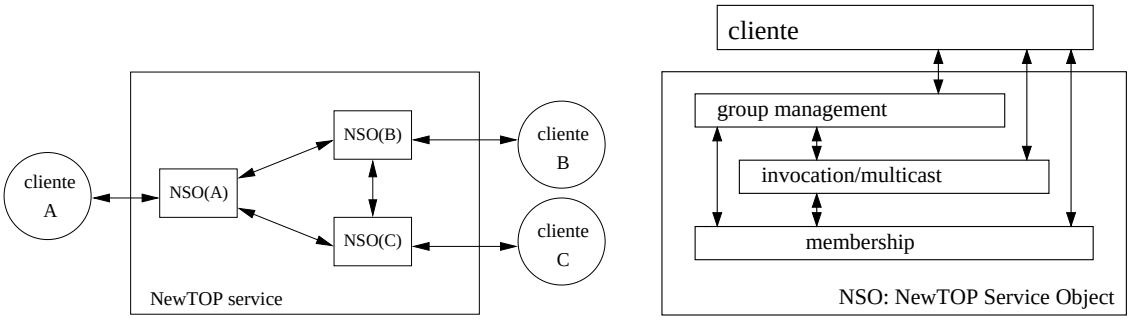
O NewTOP é um sistema de comunicação em grupo que possui os protocolos de ordenamento e controle de grupo garantindo a atomicidade na troca de mensagens. Ele assume que os membros podem estar presente em vários grupos simultaneamente e o tamanho do grupo não é limitado. Define-se que o ambiente de execução é assíncrono, onde o tempo de transmissão da mensagem não pode ser estimado com precisão. A camada de rede pode sofrer um particionamento sendo que a funcionalidade da comunicação entre os membros é preservada ([16, 11]).

O NewTOP implementa ordenamento causal e duas versões de ordenamento total: assimétrico e simétrico. No assimétrico, um único membro do grupo é responsável por determinar a ordem de entrega, enquanto no simétrico todos os membros do grupo compartilham a responsabilidade por determinar a ordem.

O controle de membros é implementado através de *groups-views.* Cada membro possui uma visão do grupo que é atualizada sempre que se detecte/suspeite a sua alteração, normalmente originada pela falha de um membro. Além disso, há garantia que todos os membros possuirão uma visão consistente do grupo, que é regida pelas propriedades

identificadas por VC - *view consistency*. Assim como as VCs, existem as propriedades chamadas de MD - *message delivery*, que garantem a atomicidade na troca de mensagens entre os membros ([11]).

Atualmente, o NEWTOP está sendo implementado via serviço CORBA ([17, 18]). Os membros dos grupos são objetos CORBA. Os clientes do serviço podem gerenciar a criação e a exclusão dos membros. Os objetos podem participar em mais de um grupo simultaneamente, permitindo também que os membros dos grupos se sobreponham. O NEWTOP é um serviço distribuído e auxiliado pelos NSOs (*NewTOP Service Object*) (ver Figura 1). Para cada cliente existe a alocação de um NSO que controla a comunicação com o grupo. A comunicação entre os NSOs é realizada pela camada ORB. A implementação dos controles oferecidos pelo NEWTOP é realizada pelos NSOs.



(a) Clientes NewTOP e seus NSOs

(b) Serviços implementados pelo NSO

Figura 1: Implementação do NEWTOP.

4 PRMP

O PRMP é um protocolo de transporte multicast baseado em *polling* que possui um mecanismo eficiente para o controle das implosões. O protocolo estende o modelo de comunicação TCP, conhecido pela maioria dos desenvolvedores de aplicações de rede, para um esquema com múltiplos destinatários. Notadamente, um protocolo de transporte multicast como o PRMP implementa controle de congestionamento multicast, de forma a permitir o compartilhamento justo dos fluxos TCP e IP multicast ([5]). A seguir, será fornecida uma breve explicação sobre o funcionamento do PRMP e os aspectos relevantes de sua implementação ([4, 5]).

Os dados são colocados em pacotes e transmitidos através de IP multicast para os receptores. O transmissor e os receptores mantém uma "janela deslizante"; o transmissor marca na sua janela quais os pacotes que foram recebidos (ACK) pelos receptores, de acordo com as respostas. Para evitar a implosão de respostas, o transmissor utiliza uma política de *polling* para controlar a quantidade de respostas geradas pelos receptores: somente quando solicitado, por uma requisição de *polling*, o receptor pode enviar, por unicast, uma resposta contendo a informação sobre ACKs e NACKs dos pacotes recebidos até o momento. Quando a resposta chega no transmissor, a sua janela é atualizada, e a detecção de perda

e recuperação podem serem executadas. O transmissor detecta a perda de pacotes através dos NACKs contidos na resposta enviada pelos receptores, após a requisição de *poll*. A recuperação é realizada através de retransmissão, que pode ser realizada por múltiplos *unicast* ou por uma única operação *multicast,* dependendo o número de cópias (do pacote) que deve ser retransmitido. As requisições de *poll* e as respostas também podem serem perdidas, e ambas as perdas são detectadas pelo transmissor através de *timeouts*. Se, o transmissor re-envia a requisição de *poll* um novo *timer* é definido para esperar por respostas; o processo é repetido até que uma resposta seja recebida ou que o transmissor retire o receptor desta sessão.

A janela do receptor é "deslizada" de acordo com os pacotes recebidos, que são consumidos pela "camada superior" (sistema de comunicação em grupo). A janela do transmissor é "deslizada" de acordo com as respostas recebidas pelos receptores, permitindo a transmissão de novos dados. Este mecanismo somente transmite novos pacotes de dados se o transmissor pode garantir que o pacote pode ser recebido e armazenado nos *buffers* dos receptores.

5 Arquitetura NewTOP + PRMP

Esta seção tece considerações sobre uma arquitetura que combinará PRMP e NewTOP em um *middleware* de grupo. O objetivo deste middleware, tal como [14], é propiciar a desenvolvedores facilidades para criar aplicações na Internet com requisitos de dependabilidade através do paradigma de comunicação em grupo.

Conforme ilustrado na Figura 2, a arquitetura é dividida em quatro camadas. A camada mais acima é a aplicação do usuário, que interage com o serviço de comunicação em grupo através de um conjunto de chamadas de método do NewTOP (conforme descrito na Seção 3). Para a aplicação, o NewTOP tem a responsabilidade de realizar a entrega atômica e ordenada de mensagens. O PRMP, logo abaixo, fornece o transporte multicast necessário para que os algoritmos do NewTOP sejam eficientemente aplicados. Por fim, a camada mais inferior corresponde aos níveis inferiores da pilha de protocolos TCP/IP, em particular UDP/IP multicast. IP multicast é necessário para a transmissão e roteamento multicast.

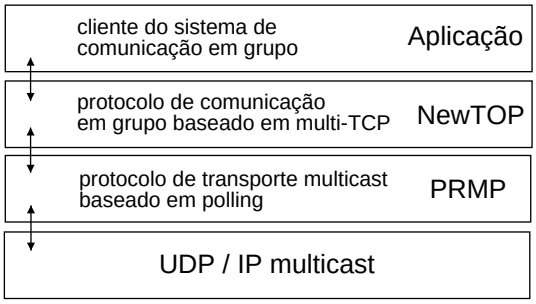


Figura 2: Arquitetura NewTOP + PRMP

A arquitetura exige a integração dos protocolos PRMP e NewTOP. NewTOP e PRMP foram projetados de forma independente. Desta forma, faz-se necessário a adaptação de

ambos os protocolos para que possam interagir e compor uma arquitetura integrada e eficiente.

O PRMP preenche os requisitos necessários para um serviço de transporte multicast como camada subjacente a um protocolo de comunicação em grupo ([3]). Estes requisitos são:

- o protocolo deve ter o controle sobre a quantidade de respostas enviadas pelos receptores para evitar a implosão;
- transmissão confiável "fim-a-fim" dos datagramas (ou bytes de um *stream*) para múltiplos receptores, detectando a perda de pacotes e recuperando-os;
- configuração e controle do grupo multicast, detectando falhas de *hosts* e particionamentos da rede;
- gerenciar os *buffers* do transmissor e receptor prevenindo a perda de pacotes desnecessária pela sobrecarga dos receptores, e
- auxiliar a prevenção de congestionamento nos gargalos da rede de uma maneira *TCP-friendly*.

Além disso, o PRMP deve:

- conhecer os membros controlados pelo NewTOP, bem como as informações geradas pelo *group-view*;detectar e informar ao NewTOP a falha de comunicação com os membros;
- repassar mensagens ao NewTOP, para que este as mantenha em uma fila até que as mesmas se tornem estáveis, e possa entregá-las à aplicação;
- lidar eficientemente com diferentes padrões de comunicação empregados pela aplicação: tráfego caracterizado pela troca de mensagens (potencialmente esporádicas) ou transmissão massiva de dados (grandes volumes).

Por fim, o NewTOP deve utilizar o PRMP para o envio das mensagens aos membros, de maneira eficiente, confirmada (ACKed) e sujeita a controle de congestionamento.

6 Considerações Finais

Os sistemas de comunicação em grupo necessitam de um protocolo de transporte multicast escalável para operarem de forma eficiente sobre a Internet. Neste artigo, considera-se uma arquitetura que combina o sistemas de comunicação em grupo NewTOP com o protocolo de transporte multicast escalável PRMP. O NewTOP oferece atomicidade, ordenamento de mensagens mais controle e gerenciamento dos membros de cada grupo. Em operações sobre a Internet, controle de congestionamento multicast deve estar presente de forma a haver um compartilhamento justo entre os fluxos TCP existentes e os fluxos gerados pelo IP multicast. O PRMP oferece a escalabilidade com controle de fluxo e congestionamento para o troca de mensagens entre os membros.

A próxima etapa é avaliar a arquitetura através de simulações utilizando o NS ([20]). Será implementado uma versão do PRMP e do NewTOP para o NS. Simulações do NewTOP utilizando multiplas conexões TCP e do NewTOP utilizando o protocolo multicast PRMP serão comparadas e avaliadas.

Agradecimentos

Agradecemos aos autores do NewTOP, em particular Graham Morgan e Dan Owen em Newcastle, pelo auxilio e discussões técnicas relativas à implementação do NewTOP.

Referências

[1] Y. Amir, et. al, "Transis: A Communication Subsystem for High Availability", FTCS-22, Boston, pp. 76-84, July 1992.

[2] Y. Amir, Danilov C., Stanton J., "A Low Latency, Loss Tolerant Architecture and Protocol for Wide Area Group Communication", FCTS-30, New York, June 2000.

[3] M. Barcellos, A. Detsch, H. Muhammad, G. Bedin, "Efficient TCP-like Multicast Support for Group Communication Systems", Brazilian Symposium on Fault-Tolerant Computing, SCTF 2001, Florianópolis, Brasil, pp. 192-206, 5-7 March 2001.

[4] M. Barcellos, P. D. Ezhilchelvan, "An End-to-End Reliable Multicast Protocol Using Polling for Scalability", In IEEE INFOCOM' 98, San Franciso, pp. 1180-1187, April 98.

[5] M. Barcellos, P. D. Ezhilchelvan, "PRMP: Poll-based Scaleable Reliable Multicast Protocol", Ph.D. Thesis, University of Newcastle, Newcastle upon Tyne, 200p., Oct. 1998.

[6] K. Birman, M. Hayden, O. Ozkasap, Z. Xiao, M. Budiu, Y. Minsky, "Bimodal multicast", ACT Transactions of Computer System, 17(2), pp. 41-88, May 1999.

[7] K. Birman, "Building Secure and Reliable Network Applications", Prentice Hall, 500p., 1996.

[8] K. Birman, A. Schiper, "Lightweight causal and atomic group multicast", ACM Transactions on Computer Systems, 9(3), pp. 272-314, August 1991.

[9] K. Birman, T. Joseph, "Reliable Communication in the Presence of Failures", Communications of ACM, 5(1), pp. 47-76, 1987.

[10] K. Chandy, L. Lamport, "Distributed Snapshots: Determining Global States of Distributed Systems", ACM Transactions on Computer Systems, 3(1), pp. 63-75, 1985.

[11] P. Ezhilchelvan, R. Macedo, S. Shrivastava, "Newtop: A Fault-Tolerant Group Communication Protocol", In IEEE 15th Intl. Conf. Distributed Computing Systems, Vancouver, pp. 296-306, May 1995.

[12] L. Lamport, "Time, clocks, and ordering of events in a distributed system", Communications of ACM, 21(7), pp. 558-565, July 1978.

[13] Lau C. L., J. Fraga, L. M. Souza, R. S. Padilha, "GroupPac: Um Framework para Implementação de Aplicações Tolerantes a Falhas", Conferência Latino Americanda de Informática, CLEI 2000, Cidade do México, México, 18-22 Setembro 2000.

[14] L. E. Moser, P. M. Melliar-Smith, "The InterGroup Protocols: Scalable Group Communication for the Internet", Globecom, Sydney, Australia, 14-16 December 1998.

[15] L. E. Moser, P. M. Melliar-Smith, "Totem: a Fault-tolerant Multicast Group Communication System", In Communications of ACM, 39(4), pp. 54-63, April 1996.

[16] G. Morgan, P. D. Ezhilchelvan, "Policies for using Replica Groups and their effectiveness over the Internet", Proceedings of the Internation Workshop on Networked Group Communication, NGC 2000, Palo Alto, California, USA, 8-10 November 2000.

[17] G. Morgan, "A Middleware Service for Fault-tolerant Group Communications", PhD. Thesis, Dept. of Computing Science, University of Newcastle upon Tyne, September 1999.

[18] G. Morgan, S. K. Shrivastava, P. D. Ezhilchelvan, M. C. Little, "Design and Implementation of a CORBA Fault-Tolerant Group Service", In 2nd IFIP WG 6.1 Internation Working Conference on Distributed Applications and Interoperable Services, Helsinki, June 99.

[19] S. Mullender, "Distributed Systems", ACM Press Frontier Series, Addison-Wesley, 580p., 1993.

[20] The network simulator - ns 2 - web site, http://www.isi.edu/nsnam/ns.

[21] K. Obraczka, "Multicast transport protocols: a survey and taxonomy", IEEE Communications Magazine, 36(1), pp. 94-102, Jan 1998.

[22] Qixiang Sun, D. Sturman, "A Gossip-based Reliable Multicast for Large-Scale High-Throughput Applications", Proceedings of the International Conference on Dependable Systems and Networks, DSN 2000, New York, 25-28 June 2000.

[23] R. Renesse, K. Birman, S. Maffeis, "Horus: A Flexible Group Communication System", Communications of ACM, 39(4), pp. 76-83, April 1996.

[24] S. Paul, K. Sabnani, J. Lin, and S. Bhattacharryya, "Reliable Multicast Transport Protocol (RMTP)", IEEE Journal on Selected Areas in Communications, 15(3), pp 407-421, April 1997.

[25] D. Towsley, J. Kurose, S. Pingali, "A Comparison of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols", IEEE Journal of Selected Areas in Communications, 15(3), pp. 398-406, 1997.