Empilhamento Dinâmico de Serviços de Comunicação em Grades de Computadores

Rodrigo Righi, Philippe Olivier Alexandre Navaux

PPGC - Programa de Pós-graduação em Computação
UFRGS - Universidade Federal do Rio Grande do Sul
{rrrighi, navaux}@inf.ufrgs.br

Resumo

As grades, ou *grids*, são dinâmicas e não confiáveis [Plateau, 2002]. Uma sentença estabelecida antes da execução pode não ser verdadeira no seu decorrer. Por exemplo, máquinas podem entrar ou sair da grade, serviços podem deixar de serem relevantes num contexto, bem como necessários noutro, e parâmetros de recursos e serviços podem mudar ou serem otimizados para uma certa ocasião [Allen et al., 2001]. Seguindo essa idéia, tem-se o plano de projetar um *middleware* para a comunicação em grades de que apresente a capacidade de adaptação. Essa última será expressa através de uma pilha de serviços de comunicação mantida dinamicamente durante a execução da aplicação. O empilhamento acontece de acordo com o contexto da grade num determinado instante e obedece a regras de execução. Pretende-se formar uma semântica de comunicação para o transporte de dados entre processos com o intuito de gerar desempenho, mantendo serviços básicos necessários (como segurança) para certos trechos da grade.

O *middleware* trata as questões de comunicação como serviços num ambiente de grade. O seu objetivo é gerir dinamicamente os serviços nas pilhas (empilhamento e desempilhamento) de modo que cada par expresse uma semântica de comunicação específica para uma transmissão de dados fim-a-fim (entre dois processos). O termo pilha foi escolhido pelo fato que cada uma de suas camadas (serviços) agrega uma semântica particular a mensagem. Para viabilizá-la, é interessante que cada serviço tenha uma interface uniforme. Assim, é possível a manutenção deles (adição e remoção) de forma arbitrária. Num primeiro momento, os serviços que formarão a pilha de comunicação estão apresentados na figura 1 (a). O assincronismo é relevante para sobrepor comunicação com computação, principalmente entre diferentes domínios. A criptografia cifra uma mensagem em trânsito pela rede. O tratamento da heterogeneidade é empilhado para a comunicação entre nós com diferentes sistemas de interconexão. O serviço ponte é relevante para comunicação entre computadores que não possuem uma conectividade direta (entre nós de diferentes agregados, por exemplo). O serviço agregador de mensagens realiza a união delas para uma posterior transmissão. O serviço de múltiplos fluxos cria várias conexões TCP/IP com o intuito de diminuir o impacto desse protocolo. Compressão de dados serve para diminuir o tamanho da mensagem e por conseqüência o tempo de comunicação. Por fim, o serviço de suporte a rede de sistema carrega os *drivers* necessários para o uso de redes rápidas.

A manutenção da pilha deve ser dinâmica e segundo um mecanismo autônomo. A dinamicidade se refere a duas características. A primeira delas é a inclusão e a remoção de serviços em tempo de execução. A segunda característica trata alterações de parâmetros

de cada serviço para atingir desempenho de acordo com as regras de execução. Dessa forma, a pilha de comunicação é totalmente reconfigurável, tanto na sua composição quanto no seu estado. A autonomia é atingida com o tratamento automático da pilha no decorrer da execução. A idéia nesse ponto é que o usuário não interfira (ou o mínimo possível) na manutenção da pilha de comunicação. A maleabilidade no tratamento dos seus serviços é disponibilizada através das regras de execução, de um mecanismo de captura de informações (contexto) sobre a grade e do serviço de monitoramento. Segundo a especificação computação autônoma [Kephart and Chess, 2003], o presente trabalho está inserido nas áreas de auto-otimização e auto-configuração.

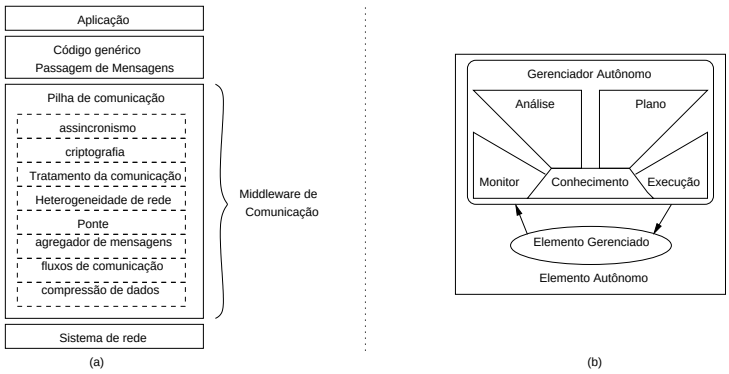

Figura 1: (a) Pilha de comunicação; (b) Elemento autônomo

A construção do *middleware* utiliza a idéia de módulos MAPE (Monitoramento, Análise, Plano e Execução) e o conceito de elemento autônomo [Kephart and Chess, 2003]. A figura 1 (b) mostra um elemento autônomo. Ele coordena uma pilha de comunicação residente num processo e um elemento gerenciado é representado pelos serviços disponíveis nessa pilha. Um processo tem uma pilha de serviços de comunicação para cada conexão que possui com os demais. Assim, num primeiro momento, a autonomia e a adaptação estariam presentes em nível da pilha de uma conexão de um processo. Por fim, uma função de avaliação de desempenho é importante para medir a eficiência de uma adaptação realizada no sistema. Por exemplo, após a troca de um parâmetro de um serviço é necessário estimar se ela alcançou os objetivos da aplicação. Por fim, serão investigados os sistemas GAT (*Grid Application Toolkit*) e LAM-MPI para as questões de implementação.

Referências

[Allen et al., 2001] Allen, G., Foster, I., Karonis, N. T., Ripeanu, M., and Toonen, B. (2001). Supporting efficient execution in heterogeneous distributed computing environments with cactus and globus. In *Supercomputing*, pages 52–52, New York, NY, USA. ACM Press.

[Kephart and Chess, 2003] Kephart, J. O. and Chess, D. M. (2003). The vision of autonomic computing. *Computer*, 36(1):41–50.

[Plateau, 2002] Plateau, B. (2002). The grid: Challenges and research issues. In *ASIAN '02: Proceedings of the7th Asian Computing Science Conference on Advances in Computing Science*, pages 13–14, London, UK. Springer-Verlag.