

Integração do Sistema AppMan de Gerenciamento de Aplicações para Ambiente de Grade com diferentes Sistemas de Gerenciamento de Recursos

Tonismar Régis Bernardo

27 de outubro de 2007

Introdução

Atualmente o uso de redes de computadores tem aumentado exponencialmente. Muitas dessas redes são distribuídas de forma geograficamente separadas precisando de uma complexa infra-estrutura de software e hardware para gerenciá-las e conectá-las. Dentre as diversas soluções existem a grade computacional (*grid computing*) possui característica viabiliza essa conexão.

Segundo Dantas [1, apud], pode-se dizer, também, que representa uma forma estendida dos serviços web permitindo que recursos computacionais possam ser compartilhados.

Defini-se grades como uma plataforma computacional heterogênia distribuída geograficamente fornecendo serviços e recursos às organizações participantes da plataforma.

[1, apud] O Global Grid Forum (GGF) uma comunidade fórum com milhares de indivíduos representando mais de 400 organizações em mais de 50 países criou e documentou especificações técnicas e experiências de usuários. O GGF definiu grades computacionais como um ambiente persistente o qual habilita aplicações para integrar instrumentos, disponibilizar informações em locais difusos. Desde lá esta não é a única e precisa definição para o conceito de grades. [2] Define um sistema em grade propondo um *checklist* de três pontos.

1. coordenar recursos quais não são direcionados para um controle central.
2. usar protocolos e interfaces padronizados, abertos para propósitos gerais.
3. oferecer QoS (qualidade de serviço) não triviais tais como: autenticação, escalonamento de tarefas, disponibilidade.

Uma definição formal do que um sistema em grade pode prover foi definido em [1, apud]. Focando na sua semântica, mostrando que grades não são apenas uma modificação de um sistema distribuído convencional. Podem apresentar recursos heterogênicos como sensores e detectores e não apenas nós computacionais. Abaixo uma lista de aspectos que evidenciam uma grade computacional [3]

- heterogeneidade
- alta dispersão geográfica

- compartilhamento (não pode ser dedicado a uma única aplicação)
- múltiplos domínios administrativos (recursos de várias instituições)
- controle distribuído

A grade deve estar preparada para lidar com todo o dinamismo e variabilidade, procurando obter a melhor performance possível adaptando-se ao cenário no momento.

Devido à grande escala, ampla distribuição e existência de múltiplos domínios administrativos, a construção de um escalonador de recursos para grades é praticamente inviável, até porque, convencer os administradores dos recursos que compõem a grade abrirem mão do controle dos seus recursos não é uma tarefa nada fácil. Escalonadores têm como características receber solicitações de vários usuários, arbitrando, portanto, entre os usuários, o uso dos recursos controlados.

[4] considera escalonar como um problema de gerenciamento de recursos. Basicamente um mecanismo ou uma política usada para, eficientemente e efetivamente, gerenciar o acesso e uso de um determinado recurso. Porém, de acordo com o GGF's [5], escalonamento é o processo de ordenar tarefas sobre os recursos computacionais e ordenar a comunicação entre as tarefas, assim sendo, ambas aplicações e sistemas devem ser escalonadas.

O gerenciamento de recursos de um sistema centralizado possui informação completa e atualizada do status dos recursos gerenciados. Este difere do sistema distribuído, o qual não tem conhecimento global de recursos dificultando assim, o gerenciamento. O ambiente em grade introduz cinco desafios para o problema de gerenciamento de recursos em ambientes distribuídos [6]:

1. autonomia: os recursos são, tipicamente propriedades e operados por diferentes organizações em diferentes domínios administrativos.
2. heterogeneidade: diferentes lugares podem usar diferentes sistemas de gerenciamento de recursos (RMS - *resource management system*).
3. estender as políticas: suporte no desenvolvimento de nova aplicação de mecanismos de gerência num domínio específico, sem necessitar de mudanças no código instalado nos domínios participantes.
4. co-alocação: algumas aplicações tem necessidades de recursos os quais só podem ser satisfeitos apenas usando recursos simultâneos com vários domínios.

5. controle online: RMSs precisam suportar negociações para adaptar necessidades de aplicações para recursos disponíveis.

Referências

- [1] P. K. V. Mangan, “Grand: Um modelo de gerenciamento hierárquico de aplicações em ambiente de computação em grade,” p. 150, 2006.
- [2] I. Foster, S. Tuecke, and C. Kesselman, “The anatomy of the grid enabling scalable virtual organizations,” p. 25, 2001. <http://www.globus.org/alliance/publications/papers/anatomy.pdf>.
- [3] W. Cirne, “Grids computacionais: Arquiteturas, tecnologias e aplicações,” p. 46, 2002. <http://walfredo.dsc.ufcg.edu.br/papers/Grids>
- [4] T. L. Casavant and J. G. Kuhl, “A taxonomy of scheduling in general-purpose distributed computing systems,” p. 37, 1996. <http://www.eng.uiowa.edu/tomc/papers/taxo.ps.gz>.
- [5] M. Roehrig, W. Ziegler, and P. Wieder, “Grid scheduling dictionary of terms and keywords,” November 2002. <http://www.ggf.org/documents/GFD.11.pdf>.
- [6] K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, and S. Tuecke, “A resource management architecture for metacomputing systems,” p. 19, 1998. <ftp://ftp.globus.org/pub/globus/papers/gram97.pdf>.