

Um modelo para busca e classificação de objetos em um sistema distribuído organizado hierarquicamente

Paulo H. Cazarotto, Cristiano A. da Costa, Rodrigo R. Righi

Programa de Pós Graduação em Computação Aplicada - UNISINOS

Av. Unisinos 950 – São Leopoldo – RS – Brasil

paulo@phcco.com, {cac, rrrighi}@unisinos.br

1. Introdução

A busca e classificação de objetos complexos em um grande volume de dados, distribuídos em várias máquinas de uma mesma rede ou na internet, possui diversas aplicações práticas cuja profundidade de busca pode ser um problema para obtenção do melhor resultado em pouco período de tempo [Wu C. et al., 2011]. Este artigo apresenta um modelo para busca e classificação de objetos organizado hierarquicamente, sua estrutura e suas operações básicas. O artigo aborda o algoritmo básico de roteamento de requisições de busca e como a indexação ocorre sobre a estrutura de dados adotada.

2. Organização básica do modelo

O modelo em desenvolvimento é baseado em uma estrutura de árvore B+Tree [Comer, 1979], na qual permite o armazenamento de objetos complexos compostos de um ou mais atributos. Estes atributos que caracterizam o objeto são utilizados para resolução de uma determinada requisição de busca de um ou mais objetos complexos. Este modelo permite que outros dispositivos (como equipamentos móveis) busquem, registrem e removam objetos em máquinas próximas ou que estão envolvidas em seu contexto.

A estrutura do modelo é estabelecida de forma hierárquica, conforme ilustra a Figura 1. A letra (a) nesta figura indica a máquina raiz, a qual pode estar associada a máquinas de roteamento ou de dados. As máquinas de roteamento de requisições (b) são responsáveis por determinada parte da estrutura e não podem armazenar objetos. Estas podem estar associadas a novas máquinas de roteamento ou máquinas de dados (c). As máquinas de dados apenas armazenam os objetos e realizam a busca em seu próprio banco de dados, enquanto as máquinas de roteamento podem repassar as requisições de busca para outras máquinas de roteamento por caminhos que já forem conhecidos (e). Por definição, os clientes apenas podem iniciar suas buscas por máquinas de roteamento que não possuem sob sua responsabilidade outras máquinas de roteamento.

O algoritmo de roteamento percorre uma aplicação distribuída da estrutura de árvore B+Tree, onde os nós folhas (máquinas de dados) são responsáveis por armazenar as informações e os nós intermediários devem armazenar os índices (máquinas de roteamento). Ao atingir uma máquina com resultados, a resposta realiza o caminho inverso. Esta ação permite que as máquinas de roteamento atualizem seus índices de acordo com as máquinas visitadas durante o processo de busca, baseando suas próximas buscas nas informações adquiridas [Guo X. et al., 2008].

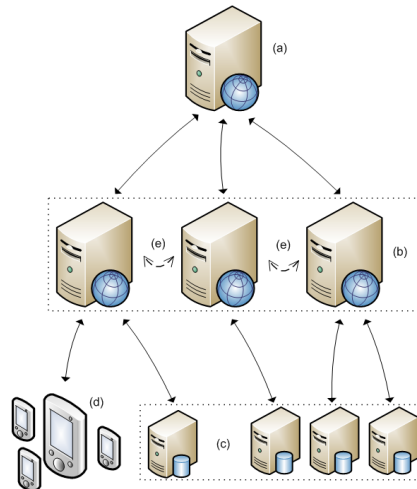


Figura 1. Hierarquia e estrutura do modelo

3. Conclusão e trabalhos futuros

O uso de uma estrutura hierárquica permite que uma subárvore se responsabilize por um conjunto de objetos relacionados em um determinado domínio. A hierarquia também organiza as máquinas participantes de forma isolar problemas e permitir a criação de ligações lógicas diretas entre nós da árvore. Ao utilizar as ligações lógicas evita-se percorrer várias máquinas até encontrar um servidor de dados com a resposta e então há um ganho no tempo de resposta.

Dentre os trabalhos futuros está o estudo de algoritmos de comparação de objetos para realização de um roteamento de busca mais eficiente, que através deste modelo pode levar em consideração o contexto e a região em que o usuário se encontra. Ainda é desejada a definição de regras para manutenção dos índices, a definição de um método de comunicação padrão para os dispositivos que enviam requisições de busca e estudos detalhados sobre a capacidade de armazenamento e processamento para verificar o comportamento das máquinas sobre determinadas situações.

Referências Bibliográficas

- Wu, C. and Wu, K. and Zeng, J. and Chen, X. (2011) "Controllable Cost Search Strategy in Unstructured P2P," *2011 Sixth Annual Chinagrid Conference*, pp. 180-187, Aug. 2011.
- Comer, D. (1979). "Ubiquitous B-Tree" In: *ACM Comput. Surv.* 11, 2 (June 1979), 121-137.
- Guo, X. and Wei, J. and Han, D. (2008) "Efficient Event Matching in Publish/subscribe: Based on Routing Destination and Matching History," *2008 International Conference on Networking, Architecture, and Storage*, pp. 129-136, Jun. 2008.