Uso de Índices e Visões Materializadas para a Otimização de Bancos de Dados

Resumo: *Atualmente, os otimizadores embutidos nos sistemas gerenciadores de bancos de dados utilizam índices e visões materializadas para produzir planos de execução de consultas otimizados. Enquanto índices e visões materializadas podem acelerar a execução de consultas, existem custos de espaço em disco e gerenciamento para mantê-los Este projeto tem como objetivo a especificação e implementação de um sistema que, dados um banco de dados, uma carga de trabalho e o espaço disponível em disco, sugira quais índices e visões materializadas devem ser criados de forma a otimizar o uso deste banco de dados*

1. Introdução

Nas últimas décadas, os Sistemas Gerenciadores de Bancos de Dados (SGBDs) têm lidado com volumes cada vez maiores de dados, bem como, com a necessidade de executar um número muito grande de consultas no menor tempo possível. Diversos pesquisadores trabalharam no desenvolvimento de novos mecanismos de otimização a fim de acelerar as consultas nestes bancos de dados.

Enquanto alguns trabalhos envolvem o re-projeto do modelo de dados de forma a armazenar mais eficientemente [Ma et al. 2007], outros desenvolveram novas heurísticas para produzir melhores planos de execução de consultas. Nos últimos dez anos, muitos SGBDs incluiram o uso de Visões Materializadas (VMs) para auxiliar a seleção dos planos de execução (ao invés da utilização apenas de tabelas convencionais e índices). Este tipo de solução aumenta o reuso de consultas pré-processadas (aquelas que geraram as VMs) para acelerar a execução das consultas.

Uma nova linha de pesquisa na área de bancos de dados é como selecionar um conjunto de índices e visões materializadas de forma a otimizar o uso dos recursos disponíveis. A seleção de visões materializadas em armazéns de dados (*data warehouses*) foi explorada por diversos pesquisadores and alguns resultados interessantes foram obtidos para bancos de dados apenas de leitura (*Read Only Databases*}) [Harizopoulos et al. 2006].

O desafio atual é o desenvolvimento de sistemas de otimização que recomende o melhor conjunto de índices e visões materializadas para bancos de dados de leitura e escrita, a fim de ajudar na seleção de planos. Esta abordagem é particularmente interessante porque ela não requer a substituição ou atualização do SGBDs que está sendo utilizado, o produto final da otimização consiste simplesmente na criação de índices e visões no SGBD em uso.

A seleção de índices e VMs é uma tarefa complexa porque deve considerar diversos tipos de restrição, como espaço em disco disponível (que é necessário para o armazenamento de índices e visões materializadas) e o custo de gerenciamento (criação e atualização) desses novos índices e VMs (sempre que uma tabela for atualizada, pode ser necessário atualizar um conjunto de índices e VMs de forma a manter a consistência do banco de dados). Desta forma, um mecanismo sofisticado de avaliação de custos e benefícios precisa ser desenvolvido.

2. Conceitos Básicos

Esta seção apresenta alguns conceitos básicos que serão utilizados por este projeto [Loney and McClain 2004, Lightstone et al. 2007, Chaudhuri 1998, Connolly and Begg 2004].

Indexação em Bancos de Dados. Todos os sistemas gerenciadores de bancos de dados atuais utilizam-se de índices para acelerar o tempo de respostas às consultas. O tipo mais simples de índice é uma lista ligada ordenada dos conteúdos de uma coluna específica de uma tabela, com ponteiros para a linha da tabela original associada ao valor do índice. Um índice permite que diversas linhas de uma tabela que satisfaçam alguma condição sejam localizadas rapidamente. Tipicamente, índices são armazenados em estruturas de dados comuns (como árvores B, hashes ou listas ligadas). Cada

índice ocupa espaço em disco e, além disso, um índice precisa ser atualizado sempre que o dado a que o índice se refere é alterado. Desta forma, um índice economiza tempo de leitura (consulta aos dados), porém aumenta o custo de inserção e atualização.

Sistema Gerenciador de Bancos de Dados Relacionais (SGBD relacionais) são aplicações para a criação e gerenciamento de bancos de dados relacionais. Nelas, índices podem ser criados ou removidos sem que a aplicação que utilize os bancos de dados precise ser atualizada. Os SGBDs decidem qual será o plano de execução que irá efetuar uma dada consulta. Esta escolha utiliza os diversos índices disponíveis e tenta otimizar o tempo de execução.

Visão Materializada. Num SGBD relacional, uma visão é uma tabela virtual que representa o resultado de alguma consulta ao bando de dados. Sempre que uma tabela de uma visão é consultada ou atualizada, o SGBD converte estas consultas ou atualizações para as tabelas subjacentes. Numa visão materializada (VM), os resultados da consulta são armazenados como tabelas concretas que podem ser atualizadas a partir das tabelas originais. Além disso, já que uma VM é armazenada como uma tabela real qualquer operação que é permitida em uma tabela normal também pode ser aplicada a uma VM.

Plano de Execução de Consultas. Um plano de execução de consultas é um conjunto de passos usados para acessar informação num banco de dados SQL de um SGBD relacional. Já que SQL é uma linguagem declarativa, há, tipicamente, diversas alternativas para se executar uma consulta, sendo que o desempenho das alternativas pode variar bastante. Quando uma consulta é submetida ao SGBD, o otimizador de consultas analisará alguns dos diferentes planos possíveis para a execução da consulta e retornará aquele que for considerado a melhor alternativa.

Otimizador de Consultas / Seleção de Plano de Execução. O tomizador de consultas é o componente do SGBD responsável por determinar qual é a maneira mais eficiente de se executar uma consulta. Ele considera alguns dos planos possíveis de execução para uma dada consulta e tenta determinar qual desses planos é o mais eficiente. O otimizador de consultas utiliza heurísticas para selecionar o melhor plano de execução.

3. Metodologia

Para o desenvolvimento deste projeto, existe um processo bem claro composto das seguintes atividades:

(i) estudo das características gerais de SGBD, que será feita com a orientação do professor;

(ii) avaliação do desempenho dos SGBDs em relação ao tempo de execução de consultas, para esta atividade pretende-se testar de três a quatro SGBDs (dentre eles o MySQL e o PostGreSQL, que são os SGBDs mais utilizados na academia);

(iii) extração de conhecimento da avaliação realizada: pretende-se usar regressão matemática para estabelecer quais fórmulas regem o comportamento dos SGBD (em relação a quantidade de registros dos bancos de dados, presença de índices, presença de visões materializadas, entre outros);

(iv) especificação e prototipação de um sistema para a sugestão de índices e visões materializadas: com base nas estapas anteriores do processo, pretende-se utilizar as fórmulas obtidas para especificar um modelo matemático do comportamento geral dos SGBDs em relação ao tamanho das tabelas, índices e visões e com isso, estimar automaticamente quais índices e visões materializadas poderiam ser criados para aumentar o desempenho do SGBD (considerando restrições de espaço em disco disponível).

4. Conclusões

Este projeto pretende ajudar no problema de otimização automática de Sistemas Gerenciadores de Bancos de Dados sugerindo a criação de índices e visões materializadas. Por se tratar de um problema extremamente complexo, não se pretende abordar todas as facetas do processo de otimização, mas sim tratar alguns dos pontos mais importantes possibilitando alcançar resultados interessantes do ponto de vista prático, científico e educacional.

A solução almejada pretende ser genérica de forma a ser aplicada em qualquer sistema gerenciador de bancos de dados relacionais que utilizem índices e visões materializadas em seus geradores de plano de execução de consultas. Além da otimização de cargas de trabalho para bancos de dados de leitura e escrita, este projeto também pretende trazer contribuições para bancos de dados apenas de leitura (como é o caso de alguns "*data warehouses*") e também para estimar o comportamento de um banco de dados na medida que o banco de dados e/ou a carga de trabalho cresçam.

5. Referências

Agrawal, S., Chaudhuri, S., and Narasayya, V. R. (2000). Automated selection of materialized views and indexes in SQL databases. In *Proceedings of 26th International Conference on Very Large Data Bases*, pages 496–505. Morgan Kaufmann.

Chaudhuri, S. (1998). An overview of query optimization in relational systems. In *Proceedings of the 17th ACM Symposium on Principles of Database Systems (PODS)*, pages 34–43. ACM Press.

Connolly, T. M. and Begg, C. E. (2004). *Database Systems: A Practical Approach to Design, Implementation and Management*. Addison-Wesley.

Goldstein, J. and Larson, P.-A. (2001). Optimizing queries using materialized views: a practical, scalable solution. In *Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data*, pages 331–342.

Harizopoulos, S., Liang, V., Abadi, D. J., and Madden, S. (2006). Performance tradeoffs in read-optimized databases. In *Proceedings of the 32nd International Conference on Very Large Data Bases (VLDB 2006)*, pages 487–498. ACM.

Lightstone, S., Teorey, T., and Nadeau, T. (2007). *Physical Database Design: the database professional's guide to exploiting indexes, views, storage, and more*. Morgan Kaufmann Press.

Loney, K. and McClain, L. (2004). *Oracle Database 10g: The Complete Reference*. McGraw-Hill Osborne Media.

Ma, H., Schewe, K.-D., and Wang, Q. (2007). A heuristic approach to cost-efficient derived horizontal fragmentation of complex value databases. In *18th Australasian Database Conference*, volume 63 of *CRPIT*, pages 103–111, Ballarat, Australia. ACS.