

Uma Ferramenta para a Sintonia de Instruções SQL*

Arlino H. M. de Araújo^{1,2}, José Maria Monteiro², José A. F. de Macêdo², Júlio A. Tavares³, Ângelo Brayner³

¹Universidade Federal do Piauí (UFPI) – Picos – PI – Brasil

²Universidade Federal do Ceará (UFC) – Fortaleza – CE – Brasil

³Universidade de Fortaleza – Fortaleza – CE – Brasil

{arlino, monteiro, jose.macedo}@lia.ufc.br, {julio, brayner}@unifor.br

Abstract. *The cost models used in current DBMS query optimizers depend on several factors, such as: statistics, estimates and properties. This fact does these optimizers, in many cases, inaccurate to pick plans, generating sub-optimal plans. In such situations, it is necessary to tune SQL statements to guide the optimizer in choosing a better execution plan than that which would normally be selected. However, this process requires a high level of expertise and skill of the database administrator. Thus, this paper presents a tool in order to support SQL statements tuning. The proposed tool uses two different approaches to support SQL tuning: assisted and automatic.*

Resumo. *Os modelos de custo utilizados pelos otimizadores de consulta dos SGBDs atuais dependem de diversos fatores que podem torná-los imprecisos, tais como: estatísticas, estimativas e propriedades. Este fato faz com que esses otimizadores sejam, em muitos casos, imprecisos na escolha de planos de execução, gerando planos sub-ótimos. Em tais situações, é necessário realizar a sintonia da instrução SQL a fim de orientar o otimizador na escolha de um plano de execução melhor do que aquele que normalmente seria selecionado. Entretanto, esse processo exige um elevado nível de especialização do administrador do banco de dados. Neste sentido, este trabalho apresenta uma ferramenta, denominada IQT (Interactive Query Tuning), que fornece suporte para o processo de sintonia de instruções SQL. O IQT possui dois modos de funcionamento distintos: assistido e automático.*

1. Introdução

Os modelos de custo utilizados pelos otimizadores de consulta atuais são bastante complexos e dependem de fatores que podem torná-los imprecisos [Bruno and Chaudhuri and Ramamurthy 2009]. Por este motivo, mesmo usando métodos de acesso e estratégias de avaliação suportadas pelo SGBD, às vezes, os otimizadores não conseguem produzir planos ótimos. Nestes casos, deve ser realizada a sintonia da instrução SQL. Para isso, em geral, duas estratégias são frequentemente utilizadas: (a) reescrever a cláusula SQL; e (b) aplicar *Query Hinting* [Shasha and Bonnet 2003].

A técnica de reescrita consiste em escrever uma nova instrução SQL equivalente à instrução SQL inicialmente utilizada (ou seja, que retorne o mesmo resultado) e que apresente ganhos de desempenho [Hamakrishnan and Gehrke 2008].

* Um vídeo de demonstração da ferramenta está disponível em <http://sbbd2013.cin.ufpe.br/screenscasts>.

Um mecanismo comum usado nos bancos de dados comerciais é denominado *Query Hinting*. Um *hint* (dica) instrui o otimizador a restringir seu espaço de busca para certo subconjunto de planos de execução (por exemplo, impondo a escolha de planos que usem um determinado tipo índice ou determinando a ordem e/ou método de junção).

O Administrador de Bancos de Dados (DBA) utiliza-se da reescrita de SQL e da inserção de *hints* para ajustar as cláusulas SQL. Entretanto, esse processo é complexo e requer conhecimentos em diferentes áreas [Herodotou and Babu 2009].

2. Solução Proposta

Este trabalho apresenta uma ferramenta, denominada IQT (*Interactive Query Tuning*), para auxiliar o processo de sintonia de instruções SQL. A ferramenta proposta utiliza duas abordagens distintas para a sintonia de instruções SQL: uma abordagem assistida e outra automática. Em ambas as abordagens um conjunto de 11 heurísticas é utilizado para realizar a sintonia (reescrita) das instruções SQL. As heurísticas são constituídas de regras para identificar oportunidades de sintonia nas instruções SQL.

A Tabela I ilustra as onze heurísticas para reescrita de instruções SQL utilizadas pela ferramenta IQT. Além disso, indica para cada heurística se esta é ou não atualmente implementada nos três principais SGBDs comerciais: PostgreSQL 8.3, Oracle 11g e SQL Server 2008.

2.1. Abordagem Assistida

A abordagem assistida consiste em um *advisor* capaz de: (i) capturar as instruções SQL anteriormente executadas, (ii) analisar essas instruções e (iii) sugerir (por meio de *alertas*, *wizards* ou relatórios) oportunidades de sintonia. Assim, o *advisor* identifica instruções SQL que se fossem reescritas poderiam fazer com que o otimizador de consultas escolhesse planos de execução melhores, reduzindo o tempo de execução das instruções SQL. Adicionalmente, permite ao DBA interagir com o processo de sintonia, por exemplo, selecionando um subconjunto das heurísticas disponibilizadas a fim de que somente essas sejam utilizadas para sintonizar as cláusulas SQL em geral ou um determinado comando SQL em particular. Se o DBA achar que algumas heurísticas são desnecessárias ou inadequadas para o seu banco ou para uma determinada consulta ele pode desativá-las.

No modo assistido, o *advisor*, periodicamente, irá analisar as instruções SQL previamente capturadas. Caso o *advisor* consiga reescrever alguma das instruções SQL analisadas, este gera um relatório contendo as recomendações de reescritas que foram aplicadas em cada uma das instruções ajustadas. Opcionalmente, o DBA pode visualizar também a lista de instruções SQL coletadas pelo *advisor*. Adicionalmente, o DBA pode fornecer um arquivo contendo um conjunto de instruções SQL a fim de que o processo de sintonia seja aplicado às instruções presentes nesse arquivo.

O DBA pode interagir com as sugestões de sintonia geradas pelo IQT. Para isso, ele deve selecionar uma das instruções SQL capturadas automaticamente pelo *advisor* ou fornecidas via arquivo e iniciar o processo de sintonia. Após a conclusão desse processo, caso alguma das heurísticas tenha sido aplicada com sucesso, a instrução SQL reescrita será exibida. O DBA pode ainda comparar a instrução SQL

original e a instrução SQL reescrita, visualizando, por exemplo, os planos de execução e os tempos de resposta gerados para cada uma delas, bem como o tempo gasto no processo de reescrita. O IQT exibe também uma descrição de todas as transformações sofridas pela instrução SQL original durante o processo de sintonia, até a geração da consulta reescrita final.

A ferramenta IQT permite ao DBA escolher e definir quais heurísticas devem ser utilizadas no processo de reescrita de uma determinada instrução SQL. Para isso, o DBA deve selecionar a instrução SQL ou digitar a instrução desejada. Em seguida, o DBA deve selecionar as heurísticas que deseja aplicar durante o processo de sintonia da instrução SQL selecionada. Além disso, o DBA pode definir um subconjunto de heurísticas a serem aplicadas às instruções SQL em geral, ou seja, para as instruções SQL que não possuam definições específicas. Vale destacar que tanto as configurações definidas para uma instrução SQL em particular quanto para as instruções SQL em geral podem ser utilizadas pelo *Middleware* que compõe a abordagem automática.

2.2. Abordagem Automática

A abordagem automática consiste em um *middleware* que atua entre a aplicação e o SGBD. Este *middleware* é responsável por: (i) receber as cláusulas SQL enviadas pelas aplicações; (ii) analisar e ajustar (reescrever) as cláusulas SQL recebidas (se necessário); (iii) enviar as cláusulas (reescritas ou não) para o SGBD e (iv) receber do SGBD o resultado da execução do comando SQL e enviá-lo à aplicação.

O *middleware* desenvolvido consiste em uma classe Java que pode ser utilizada pelos desenvolvedores de aplicação a fim de que uma determinada instrução SQL seja reescrita antes de ser enviada ao SGBD. Assim, os desenvolvedores não mais utilizariam a API JDBC (pacote java.sql) diretamente, e sim por meio do *middleware*. Assim, antes de executar uma determinada instrução SQL, o *middleware* verifica se a instrução possui oportunidades de sintonia e a reescreve (se necessário). Após esta etapa, a instrução SQL (reescrita ou original) é enviada ao SGBD. Após executar a instrução SQL, os resultados são retornados para a aplicação. Opcionalmente, o *middleware* pode checar se o custo de execução da instrução SQL original é menor que o da instrução reescrita, com a finalidade de garantir que o tempo de execução da instrução reescrita seja menor que o da original.

As abordagens assistida e automática apresentam ainda as seguintes características:

Não-intrusiva: completamente desacoplada do código do SGBD. Isso permite que a solução concebida possa ser utilizada com qualquer SGBD. Além disso, a solução não está sujeita a modificações em seu código a cada nova versão do SGBD.

Independente de localização: pode executar em uma máquina distinta daquela utilizada para hospedar o SGBD, não consumindo recursos do servidor onde o SGBD está hospedado.

3. Trabalhos Relacionados

Em [Herodotou and Babu 2009] é apresentada uma ferramenta denominada *zTuned* cuja finalidade é facilitar experimentos relacionados à sintonia de consultas SQL. A ferramenta produz conjuntos de planos que possuem operadores com a

mesma cardinalidade (chamados de planos da vizinhança) e escolhe o plano ótimo entre os melhores planos de cada vizinhança, utilizando-se do mecanismo de estimativa de custos do próprio SGBD. No Oracle 10g é introduzido um componente denominado *Automatic SQL Tuning Advisor* que fornece recomendações de oportunidades de reescrita de consultas utilizando equivalências semânticas [Dageville and Dias 2006].

Existem algumas ferramentas tais como o *IBM Optim Development Studio* [Studio 2010], o *Embarcadero DB Optimizer XE* [Optimizer 2010] e o *Quest SQL Optimizer for Oracle* [Oracle 2010] que já realizam sintonia de consultas através de recomendações de reescrita, de uso de *hints* e/ou de criação de índices. Entretanto, elas adotam uma abordagem *offline* na solução do problema e transferem para o DBA a tarefa de capturar e fornecer a carga de trabalho, dentre outras.

4. Resultados Experimentais

A fim de avaliar a eficácia da ferramenta IQT utilizando as abordagens assistida e automática investigamos dois cenários distintos. No primeiro cenário utilizamos o *benchmark* TPC-H, o qual consiste um *benchmark* voltado para aplicações de suporte à decisão. O TPC-H possui um conjunto de 23 consultas *ad-hoc*. No segundo cenário utilizamos a base de dados do *benchmark* TPC-H e uma carga de trabalho sintética formada por 30 consultas (com problemas de sintonia).

Para cada cenário, três testes foram executados: i) executou-se a carga de trabalho contendo as consultas originais (sem reescrita), o que será utilizado como *baseline*; ii) a carga de trabalho original (sem reescrita) foi submetida ao *advisor* (abordagem assistida) e, em seguida, a nova carga de trabalho (contendo as consultas ajustadas) foi executada; e iii) cada consulta da carga de trabalho original foi enviada para o *middleware* (abordagem automática com todas as 11 heurísticas habilitadas), o qual procedeu à reescrita da consulta (quando necessário) e, em seguida, executou a consulta reescrita. Para cada teste executamos 1, 2, 4, 8, 16 e 32 iterações da carga de trabalho (ou seja, 1, 2, 4, 8, 16 e 32 execuções das 23 consultas do TPC-H).

Em cada teste, as cargas de trabalho foram submetidas de três formas distintas: sequencial, aleatória, aleatória fixa (onde define-se uma sequência aleatória e depois mantém-se essa sequência). Devido às limitações de espaço iremos discutir somente os resultados dos testes sequenciais. O ambiente de execução foi composto por uma estação Core i3-2100 3.10GHz, com 4GB de RAM e 500 GB de HD. Os SGBDs utilizados foram PostgreSQL 8.3, Oracle 11g e SQL Server 2008.

4.1. Cenário 1: Benchmark TPC-H

Das 23 consultas que compõem o *benchmark* TPC-H, duas (consultas 18 e 20) foram reescritas (sintonizadas) pelo IQT, por meio da heurística H8, para os SGBDs SQL Server e PostgreSQL. Observando as Figuras 1 e 2 podemos concluir que a abordagem assistida apresentou uma pequena diminuição no tempo de execução da carga de trabalho. O que é explicado pelo fato de somente duas consultas do TPC-H terem apresentado oportunidades de sintonia. Porém, essas duas consultas não foram reescritas pelo IQT para o Oracle, uma vez que este já implementa essa heurística (H8). Assim, observamos na Figura 3 que a abordagem automática apresentou uma leve piora em relação ao *baseline*. O que é explicado pelo fato da abordagem

automática ter tido o *overhead* de tentar reescrever cada uma das consultas recebidas e nenhuma delas possuir oportunidade de sintonia no Oracle.

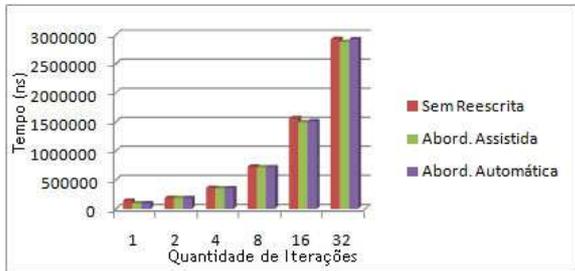


Fig. 1. Benchmark TPC-H no PostgreSQL

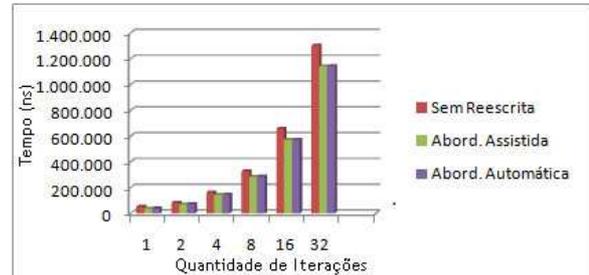


Fig. 2. Benchmark TPC-H no SQL Server

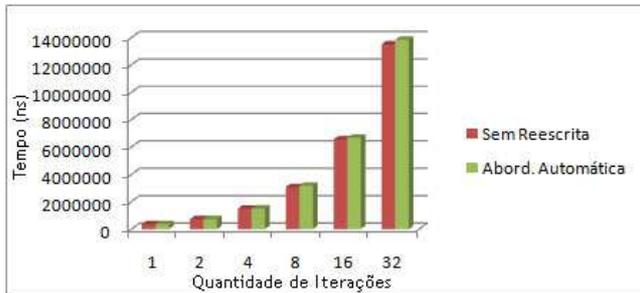


Fig. 3. Benchmark TPC-H no Oracle

4.2. Cenário 2: Base do TPC-H + Carga de Trabalho Sintética

As Figuras 4, 5 e 6 mostram que a abordagem assistida proporcionou uma grande redução no tempo de execução das cargas de trabalho. Já a abordagem automática proporcionou benefícios menores uma vez que esta envolve o *overhead* de sintonizar as instruções SQL recebidas em tempo de execução.

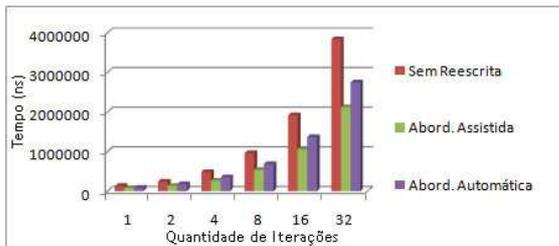


Fig. 4. Consultas sintéticas na base TPC-H no PostgreSQL

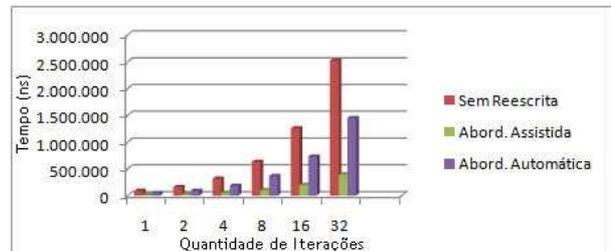


Fig. 5. Consultas sintéticas na base TPC-H no SQL Server

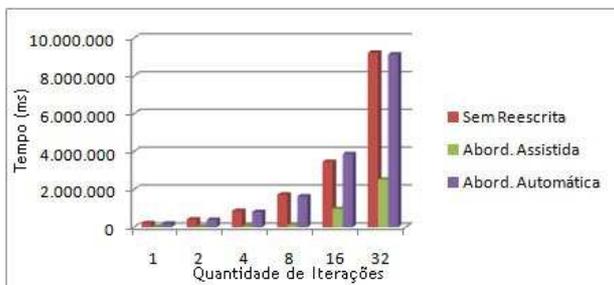


Fig. 6. Consultas sintéticas na base TPC-H no Oracle

5. Conclusões

Neste trabalho apresentamos uma ferramenta, denominada IQT, cujo objetivo consiste em auxiliar o processo de sintonia de instruções SQL em bancos de dados relacionais. A ferramenta proposta utiliza duas abordagens distintas para a sintonia de instruções SQL: uma abordagem assistida e outra automática. A ferramenta IQT pode ser utilizada em situações onde o otimizador de consultas não consegue produzir planos ótimos, mesmo usando métodos de acesso e estratégias de avaliação suportadas pelo SGBD. Os resultados dos experimentos realizados indicam que as duas abordagens utilizadas pela ferramenta IQT podem proporcionar ganhos de desempenho.

Referências

- Boulic, R. and Renault, O. (1991) “3D Hierarchies for Animation”, In: *New Trends in Animation and Visualization*, Edited by Nadia Magnenat-Thalmann and Daniel Thalmann, John Wiley & Sons Ltd., England.
- Bruno, N., Chaudhuri, S. and Ramamurthy, R. (2009) “Power Hints for Query Optimization”, In: *Proceedings of the 2009 IEEE International Conference on Data Engineering*, ACM, Washington, DC, USA.
- Dageville, B. and Dias, K. (2006) “Oracle’s Self-Tuning Architecture and Solutions”, In: *IEEE Computer Society Technical Committee on Data Engineering*, Oracle, USA.
- Dyer, S., Martin, J. and Zulauf, J. (1995) “Motion Capture White Paper”, http://reality.sgi.com/employees/jam_sb/mocap/MoCapWP_v2.0.html, December.
- Hamakrishnan, R. and Gehrke, J. (2008) “Sistemas de Bancos de Dados”, São Paulo: McGraw-Hill.
- Herodotou, H. and Babu, S. (2009) “Automated SQL Tuning through Trial and (Sometimes) Error”, In: *DBTest’09*, ACM, Rhode Island, USA.
- Holton, M. and Alexander, S. (1995) “Soft Cellular Modeling: A Technique for the Simulation of Non-rigid Materials”, *Computer Graphics: Developments in Virtual Environments*, R. A. Earnshaw and J. A. Vince, England, Academic Press, p. 449-460.
- Knuth, D. E. (1984), *The TeXbook*, Addison Wesley, 15th edition.
- Krishnaprasad, M., Liu, Z. H., Manikutty, A., Warner, J. W., Arora, V. and Kotsovolos, S. (2004) “Query Rewrite for XML in Oracle XML DB”, In: *Proceedings of the Vldb Conference*, Toronto, Canada.
- Moro, M. M. *The Role of Structural Aggregation for Query Processing over XML Data*. Ph.D. thesis, University of California, Riverside (UCR), USA, 2007.
- Optimizer, D. X. (2010) “Embarcadero DB Optimizer XE”, Available: <http://www.embarcadero.com/products/db-optimizer-xe>.
- Oracle, S. O. (2010) “Quest SQL Optimizer for Oracle”, Available: <http://www.quest.com/SQL-Optimizer-for-Oracle>, June.
- Shasha, D. and Bonnet, P. (2003) “Database Tuning: Principles, Experiments and Troubleshooting Techniques”, Morgan Kaufmann.
- Smith, A. and Jones, B. (1999). “On the complexity of computing.” In *Advances in Computer Science*, pages 555–566. Publishing Press.
- Studio, O. D. (2010) “IBM Optim Development Studio”, Available: <http://www-01.ibm.com/software/data/optim/development-studio>.