

AVALIAÇÃO DE DESEMPENHO DE FERRAMENTAS PARA *TUNING* EM BANCO DE DADOS¹

EVALUATION OF TOOL PERFORMANCE FOR TUNING IN DATABASE

Tadeu Pesamosca Coradini² e Gustavo Stangherlin Cantarelli³

RESUMO

A informação digital gerada, atualmente, alcança uma quantidade cada vez maior. Neste sentido, é fundamental gerenciar e otimizar as bases de dados que armazenam informações na constante busca por melhorias de desempenho no acesso a tais informações. Para isso, foram testadas as ferramentas de auxílio ao ajuste de desempenho – *tuning* – dos principais Sistemas Gerenciadores de Banco de Dados utilizados (SQL Server, Oracle e PostgreSQL), utilizando consultas SQL pré-estabelecidas como carga para avaliar a capacidade das ferramentas desses sistemas quanto a sugerir melhorias de processamento que trouxessem um ganho de desempenho significativo.

Palavras-chave: ajuste de desempenho, sistemas gerenciadores, Oracle, SQL Server, PostgreSQL.

ABSTRACT

The generation of digital information is each time bigger. In this sense, it is essential to manage and optimize databases in the constant search for performance improvement in the access to such information. For this purpose, it was tested some tools that aid performance tuning of the main Management Database Systems used (SQL Server, Oracle and PostgreSQL). Predetermined SQL queries were used to evaluate the ability of these systems to suggest processing improvements that may bring significant performance gain.

Keywords: *performance tuning, management Systems, Oracle, SQL Server, PostgreSQL.*

¹ Trabalho de Final de Graduação - TFG.

² Acadêmico do Curso de Sistemas de Informação - Centro Universitário Franciscano. E-mail: tcoradini@hotmail.com

³ Orientador - Centro Universitário Franciscano. E-mail: gus.cant@gmail.com

INTRODUÇÃO

Um Sistema Gerenciador de Banco de Dados (SGBD), segundo (MULLINS, 2010), é um conjunto de *softwares* desenhado para criar, projetar e gerenciar banco de dados, permitindo o compartilhamento de informações por seus usuários, tendo como seu principal objetivo recuperar e disponibilizar estes dados com a maior eficácia possível. No entanto, com o passar do tempo e o crescimento da informação, a perda de desempenho se torna um fator preocupante na administração da base de dados. Para tentar resolver esse problema, é empregada a técnica de *tuning* ou sintonia, que de acordo com Lifschitz et al. (2004), pode ser descrita como uma técnica de ajustar parâmetros, configurações e estruturas de um sistema de banco de dados de acordo com as exigências de uma determinada carga de trabalho.

Em seu trabalho, Parra (2005) destaca algumas técnicas de *tuning* utilizadas para um estudo comparativo de desempenho em uma plataforma Oracle. Foram considerados alguns quesitos que afetam diretamente o desempenho do sistema, de forma positiva ou não, como os ajustes na configuração de *hardware*, parâmetros de configuração do próprio SGBD (memória, armazenamento, distribuição de dados), refinamento do projeto do banco de dados, ajuste de SQL.

Em um experimento prático utilizando PostgreSQL, Telles (2009) simula o comportamento do SGBD no modo normal de operação e no modo *tuning*, onde parâmetros de configuração somente do SGBD são alterados, operações de gravação, exclusão, alteração e consultas são realizadas de forma a simular a operação em ambiente de produção e os efeitos dessas alterações no comportamento do banco de dados.

O foco deste trabalho foi direcionado à questão de problemas de elaboração e ajuste de consultas dos SGBD, analisando as ferramentas disponíveis nos próprios Sistemas Gerenciadores, verificando como elas podem auxiliar na melhoria do desempenho do Banco de Dados na execução das consultas.

REFERENCIAL TEÓRICO

TUNING DE DESEMPENHO

Segundo Date (2004), para se alcançar o desempenho desejado em um banco de dados, o ajuste de parâmetros é uma exigência necessária.

Os objetivos do *tuning*, de acordo com Elmasri e Navathe (2005), são que as aplicações sejam executadas mais rapidamente, a diminuição do tempo de resposta de consultas/transações e a melhora do desempenho geral das transações. Em relação à medição do desempenho, Silberschatz et al. (1999) afirmam que é possível medi-la de acordo com a teoria da oferta e demanda, ou seja, o desempenho de um banco de dados é a taxa entre os pedidos que o SGBD recebe e a demanda para informação.

Ainda, existem 5 fatores que influenciam o desempenho do banco de dados:

- *Workload* - define a demanda do banco de dados;
- *Throughput* - capacidade do computador de processar dados;
- Recursos do Sistema - são o *hardware* e as ferramentas de *software* disponíveis no sistema;
- Otimização - otimização de pesquisas realizada internamente ao SGBD;
- Concorrência - condição em que dois ou mais componentes do *workload* estão tentando o mesmo recurso em modos conflitantes.

Sendo assim, o ponto chave da utilização de um SGBD se refere ao seu desempenho, uma vez com problemas de desempenho o sistema perderá sua confiabilidade. Nesse sentido, é importante ter um plano de monitoramento e ajuste do banco de dados, a fim de evitar sua degradação (IKEMATU, 2002).

Para a busca do *tuning* de desempenho, Silberschatz et al. (2006) destacam alguns passos a serem seguidos como:

- **Localização de Gargalo:** Gargalo em um sistema é um ou vários componentes que limitam o desempenho global do sistema;

- **Parâmetros Ajustáveis:** Sistemas de banco de dados podem ser ajustados em três níveis: Nível mais baixo – Alterações de *Hardware*. Segundo nível – Modificações de parâmetros do sistema de banco de dados. Nível mais alto – Ajuste no projeto do esquema e transações, criação de índices de pesquisa;

- **Ajuste de Hardware:** Garante a robustez do subsistema de disco e da memória utilizada para armazenar todas as páginas;

- **Ajuste de Esquema:** Modificações no esquema de relações entre tabelas, alterando a forma como as mesmas são acessadas;

- **Ajuste de Índices:** Criação de índices de referência sobre as relações para que se consiga uma melhora no desempenho;

- **Visões Materializadas:** Armazenam um conteúdo calculado, facilitando e agilizando a requisição desta consulta;

- **Ajuste Automatizado do Projeto Físico:** Realizado por ferramentas presentes em sistemas de banco de dados comerciais com o objetivo de criação de índices, visão materializada, particionamento de tabelas;

- **Ajuste de Transações:** Utiliza a técnica da melhoria da orientação do conjunto, que tenta minimizar a alta sobrecarga de comunicação no envio de múltiplas consultas SQL utilizando a SQL embutida;

- **Simulação de Desempenho:** Utilizada pelo administrador do banco de dados (DBA) para modelar alguns aspectos de cada serviço ativo, com o objetivo de analisar o comportamento do sistema sobre várias condições de carga de trabalho e tempo de serviço.

FERRAMENTAS DE *TUNING*

Atualmente, existem diversas ferramentas para auxílio à técnica de *tuning* em banco de dados, geralmente embutidas em sistemas gerenciadores comerciais mais famosos. Neste trabalho, foram utilizadas apenas ferramentas nativas dos sistemas gerenciadores, visto que o foco foi avaliar somente o desempenho da utilização das mesmas em conjunto com seu SGBD de origem, excluindo dessa forma programas de *benchmark* de terceiros.

Performance Monitor

Ferramenta gráfica do Microsoft SQL Server 2008 que monitora diferentes atividades no sistema. Dividido em monitor do sistema/*logs* e alertas de desempenho. Esse monitor provê estatísticas e armazena *logs* de atividades no sistema. Essa informação recolhida auxilia na verificação de problemas de desempenho.

SQL Server Profiler

Utilitário presente no Microsoft SQL Server 2008 permite o monitoramento e registro de atividades do banco de dados em um determinado período de tempo e, ainda, a exibição de qualquer instrução SQL enviada a uma instância do SQL Server (SILBERSCHATZ et al., 2006).

Database Engine Tuning Advisor (DTA)

Presente no SGBD da Microsoft, tem a função de projetar índices de acesso, *views* materializadas e particionamento de tabelas utilizando os dados coletados previamente pelo SQL Server Profiler para sugerir mudanças e alterações de parâmetros no banco de dados (SILBERSCHATZ et al., 2006).

SQL Tuning Advisor

Ferramenta de auxílio a *tuning* do SGBD Oracle 11g. Ela pode ser utilizada para gerar planos de execução mais eficientes, sendo carregada com dados de instruções de alta carga monitoradas pelo Oracle. Essas instruções, por utilizarem muitos recursos, sofrem recomendações de melhorias no que diz respeito a algumas categorias como:

- Análise de estatísticas - são informações necessárias ao otimizador para avaliar a melhor forma de execução;

- Profiling SQL - conjunto de informações com a utilidade de auxiliar o otimizador a tomar decisões melhores da próxima vez que a instrução for executada;
- Análise do caminho de Acesso - análise do otimizador sobre o caminho de acesso dos dados, sugerindo a criação de índices;
- Análise da estrutura SQL - o otimizador analisa a estrutura SQL e sugere alterações no intuito de uma execução mais eficiente (SILBERSCHATZ et al., 2006).

Automatic SQL Tuning

O *tuning* automático do Oracle é um recurso introduzido na versão 11g que utiliza o SQL Tuning Advisor em janelas de manutenção programada sobre as instruções SQL para criar *jobs* automáticos de ajuste das instruções com baixo desempenho e alto consumo de recursos armazenados historicamente no Repositório Automático de Carga de Trabalho (AWR).

Esse ajuste realizado na instrução gera um perfil que é avaliado como aceitável ou não em relação ao novo desempenho (deve ser pelo menos 3 vezes melhor que o anterior). Se constatada essa melhora, o perfil novo da instrução é aplicado e será executado nas próximas ocorrências da instrução SQL.

Diferentemente das técnicas de ajuste menos automatizadas vistas em Burgo (2007), o *tuning* automático do Oracle surge para auxiliar ou até realizar ajustes conforme sua análise da condição de necessidade do sistema.

SQL Profiler

Presente no Postgres Enterprise Manager, esta ferramenta funciona capturando e exibindo uma carga de trabalho para posterior análise em um rastreamento SQL. As consultas rastreadas podem ser salvas para uma análise posterior (POSTGRES ENTERPRISE MANAGER GETTING STARTED GUIDE, 2011).

Index Advisor

Conselheiro de índices do PostgreSQL, examina as consultas rastreadas pelo SQL Profiler e faz recomendações de criação de índices, quando for possível, em todas as tabelas adjacentes para melhorar o tempo de resposta (POSTGRES ENTERPRISE MANAGER GETTING STARTED GUIDE, 2011).

METODOLOGIA

Para a avaliação das ferramentas de auxílio a *tuning* em banco de dados, foram escolhidos testes que forçaram as ferramentas a propor algum tipo de melhoria na consulta que estava sendo avaliada.

A base de dados utilizada no teste é do Sistema de Estatísticas da SUSEP, atualizada até 04/2011 e continha 3.944.390 registros divididos em 11 tabelas, conforme especificado na tabela 1.

Tabela 1 - Nome das tabelas de teste e número de registros das mesmas.

Nome da tabela	Quantidade de registros
auto2_grupo	475
auto2_vei	4745
auto_sexo	4
auto_cat	8
auto_cidade	3439
auto_idade	6
auto_reg	41
Premreg	205
Sinreg	200
arq_casco_comp	1681807
arq_casco4_comp	2253460

O objetivo desses testes foi de demonstrar como as ferramentas de otimização iriam trabalhar com consultas simples e complexas, não otimizadas, verificando se as mesmas seriam capazes de orientar e sugerir recomendações para aumentar o desempenho de consulta. Os testes de cada SGBD foram executados isoladamente, de forma que somente o SGBD executado estaria ativo no sistema. A instalação padrão dos SGBD ocorreu no ambiente operacional Windows Seven Professional SP1, com sistema de arquivos NTFS em uma plataforma PC mono-processada. A configuração da máquina de testes utiliza uma CPU AMD 64 X2 TK-53 rodando em frequência de 1.70 GHz com 2GB DDR 2 de memória, disco rígido de 7200RPM com capacidade de 120 GB.

Os testes compreenderam a execução de consultas (*selects*), cada uma executada 4 vezes, sendo a primeira para carregar o plano de execução na memória e as outras 3 em sequência. O gerador de consultas utilizado foi o nativo de cada SGBD.

O ponto mais importante da avaliação foi verificar se as ferramentas de auxílio a *tuning* contribuíram de forma eficaz para a melhoria no desempenho das consultas, ou propuseram algum tipo de melhoria significativa, desse modo ficando comprovado ou não sua eficiência.

DESCRIÇÃO DA EXECUÇÃO DOS TESTES

O primeiro passo para execução dos testes de avaliação foi a migração da base de dados, originalmente no formato .mdf do Access para o Oracle, SQL Server e PostgreSQL. Nesse ponto, surgiram problemas relacionados à conversão de tipo de arquivos e modelagem de dados, sendo que alguns registros não eram referenciados e tiveram de ser eliminados. Para resolver esse problema de migração, foi desenvolvida uma ferramenta de migração na linguagem Java, que fazia a inserção correta dos dados nas novas tabelas de cada SGBD.

Quanto as suas funcionalidades, é importante notar que somente o SGBD Oracle tem o poder

de recomendação de reescrita de consulta, os outros SGBD estão restritos à implementação de índices, visões materializadas ou ao particionamento de tabelas.

Uma das métricas utilizadas nesta avaliação foi a medida de custo, que é responsável pela avaliação do gasto computacional para a execução de uma tarefa, nesse caso a geração de um plano de consulta. É importante ressaltar que devido a não existir um padrão global acordado para definição de um cálculo de custo comum a todos os bancos de dados, essa métrica não foi levada em consideração, somente constando como fonte de verificação de resultados pré e pós *tuning*. De acordo com Gorges (2008), a lógica empregada no cálculo do gasto computacional não é documentada e altera-se conforme as versões dos SGBD. O Oracle gera estimativas de custo baseado nas estatísticas contidas no dicionário de dados e nos recursos existentes na máquina. Já no SQL Server, o custo é definido com base no número total de linhas processadas em cada nível de um plano de consulta e no modelo de custo do algoritmo ditado pelos operadores usados na consulta. Já o PostgreSQL avalia os custos através do registro de estatísticas sobre a distribuição dos dados dentro da tabela.

A métrica a ser avaliada como resultado de comparação de desempenho é o tempo de resposta das consultas retornadas, além do aperfeiçoamento final conseguido. Inicialmente, são mostrados os resultados de cada uma das consultas sem realizar o ajuste e, posteriormente, se as ferramentas detectaram algum tipo de orientação ou conselho para a melhoria de desempenho das mesmas.

Os resultados da execução das consultas em relação ao tempo de resposta antes do *tuning* (pré-*tuning*) e depois da utilização desta técnica (pós-*tuning*) estão detalhadas nas tabelas 1, 2, 3 e 4.

Consulta 1: A consulta foi direcionada somente a uma tabela, utilizando a função *sum* (somatório), um filtro *between* e uma função de agrupamento *group by*:

```
SELECT SUM(ACP.exposicao + (ACP.exposicao * 0.05)) as TOTAL FROM arq_casco4_comp as ACP WHERE ACP.ano_modelo between 1950 and 1960 group by ACP.ano_modelo;
```

Total de 11 linhas selecionadas.

Tabela 2 - Valores de resposta, custo e aperfeiçoamento na consulta 1.

	SQL Server	PostgreSQL	Oracle
Pré-tuning			
Resposta 1 (ms)	480	2330	5690
Resposta 2 (ms)	462	2747	5760
Resposta 3 (ms)	472	2223	6090
Custo	35.989	4247.27	6.381
Pós-tuning			
Aperfeiçoamento (%)	99.0	94.0	97.0
Resposta (ms)	87	31	4
Custo	0.0034	72.24	143

Nessa primeira consulta, as orientações das ferramentas de auxílio a *tuning* dos SGBD aconselharam a criação de índices para a otimização ao acesso a dados. Nesse caso, os índices implementados mostraram uma real melhoria tanto no tempo quanto no custo de execução da *query* (consulta),

como pode ser constatado na tabela 2 nas condições pós *tuning*.

Consulta 2: A consulta abaixo foi direcionada para as tabelas **auto2_grupo** e **auto2_vei** sendo simulada uma condição de busca sem um filtro adequado, gerando dessa forma um produto cartesiano. Essa prática deve ser evitada, pois sobrecarrega demais o SGBD, tem um alto custo e produz um grande volume de dados.

```
SELECT auto2_grupo.*, auto2_vei.descricao FROM auto2_vei, auto2_grupo;
```

Total de 2.253.875 linhas selecionadas.

Tabela 3 - Valores de reposta, custo e aperfeiçoamento na consulta 2.

	SQL Server	PostgreSQL	Oracle
Pré-tuning			
Resposta 1 (ms)	50694	53527	28550
Resposta 2 (ms)	50395	38629	26150
Resposta 3 (ms)	51541	34233	26090
Custo	10.015	70753.68	3.899
Pós-tuning			
Aperfeiçoamento (%)	-	-	-
Resposta (ms)	-	-	-
Custo	-	-	-

A segunda consulta diferenciou-se da primeira, pois, por não ter a capacidade de orientação à reescrita de consultas, as ferramentas dos SGBD PostgreSQL e SQL Server não conseguiram estipular qualquer orientação de melhoria. Nesse ponto, a vantagem foi toda do Oracle, pois recomendou que houvesse a remoção da tabela ou *view* dessa consulta ou que fosse adicionada alguma condição de restrição para a seleção de dados.

Consulta 3: A consulta lista todos os registros da tabela **arq_casco_comp** cujo código das idades relacionadas na tabela **auto_idade** fossem diferentes de 0 ou de 5, ordenados pelo código. Nesse caso, utilizou-se o operador *not in* para limitar a consulta, porém, esse operador é tido como de comportamento lento e de baixo desempenho, não sendo uma boa prática de utilização.

```
SELECT * FROM arq_casco_comp WHERE idade not in (SELECT codigo FROM auto_
idade WHERE codigo = 0 or codigo = 5) order by codigo;
```

Total de 1.095.450 linhas selecionadas.

Ao analisar o resultado da tabela 4, foi possível verificar que o SGBD Oracle teve uma pequena vantagem em relação ao tempo de execução sobre o SQL Server. Porém, o maior ganho que a ferramenta de *tuning* do Oracle proporcionou foi a possibilidade de acesso a outros planos de execução (inclusive em termos de custo melhores que o original) dando outras opções de escolha ao administrador do banco de dados.

Tabela 4 - Valores de reposta, custo e aperfeiçoamento na consulta 3.

	SQL Server	PostgreSQL	Oracle
Pré-tuning			
Resposta 1 (ms)	53490	93934	47720
Resposta 2 (ms)	50189	82040	48830
Resposta 3 (ms)	49326	96253	48610
Custo	211.957	82152.13	21.439
Pós-tuning			
Aperfeiçoamento (%)	-	-	-
Resposta (ms)	-	-	-
Custo	-	-	-

Consulta 4: A consulta avaliou a utilização de uma cadeia de *left joins* entre as tabelas **arq_casco4_comp**, **auto_cat**, **auto_reg**, **auto2_vei**, **auto2_grupo** e **arq_casco_comp**, com o objetivo de recuperar os valores da coluna exposição e premio da tabela **arq_casco4_comp**.

SELECT ACC.exposicao, ACC.premio FROM arq_casco4_comp as ACC left join auto_cat as AC on (ACC.cod_tarif = AC.codigo) left join auto_reg as AR on (ACC.regiao = AR.codigo) left join auto2_vei as AV on (ACC.cod_modelo = AV.codigo) left join auto2_grupo as AG on (AV.cod_grupo = AG.grpid) left join arq_casco_comp as ACCC on (AC.codigo = ACCC.cod_tarif) WHERE AC.categoria is not NULL and AR.descricao like 'RS%' and AG.descricao like 'AUDI TT' and ACCC.premio1 > 0;

Total de 3.066.557 linhas selecionadas.

Tabela 5 - Valores de reposta, custo e aperfeiçoamento na consulta 4.

	SQL Server	PostgreSQL	Oracle
Pré-tuning			
Resposta 1 (ms)	69571	39633	49540
Resposta 2 (ms)	56009	44041	45860
Resposta 3 (ms)	56253	39131	42460
Custo	108.946	446029.34	12.216
Pós-tuning			
Aperfeiçoamento (%)	59.0	12.0	1.0
Resposta (ms)	70414	34935	46910
Custo	43.743	397732.77	12.314

É possível notar que, nessa consulta, as orientações das ferramentas de ajuste tiveram resultados bastante diferentes. Enquanto houve uma melhoria de 12,0% no custo através de dois índices implementados no PostgreSQL e aproximadamente 1,0% de melhoria no tempo de execução do plano alternativo do Oracle, no SQL Server (embora a ferramenta estimasse um aperfeiçoamento de 59%), houve uma piora no tempo de execução através do particionamento de tabelas e a criação de três índices.

Com o resultado dessas consultas de avaliação de desempenho das ferramentas de auxílio a *tuning* em banco de dados, fica claro que essas ferramentas, mesmo tendo certo grau de automatização na avaliação da otimização, necessitam de um acompanhamento e de um parecer final do admi-

nistrador do banco de dados, visto que ainda não tem uma capacidade de avaliação baseada em bom senso e na análise futura do impacto gerado por suas escolhas.

CONCLUSÃO

O resultado do presente trabalho mostra que o processo de *tuning* de banco de dados é fundamental para a manutenção de um sistema, de maneira que ele consiga atender os requisitos de funcionamento com o melhor desempenho possível. Além disso, é possível ter uma noção da importância da utilização de ferramentas que auxiliem o administrador da base de dados em sua contínua tarefa de ajuste. Através da execução dos testes, pela gama de recursos disponibilizados e do desempenho das ferramentas de cada SGBD ficou claro a superioridade do sistema da Oracle em relação a seus concorrentes, devido às tecnologias inovadoras utilizadas e seu foco na otimização de desempenho com grandes massas de dados. Em seguida está o sistema da Microsoft que também oferece boas ferramentas, porém, não tão interativas e “inteligentes” quanto às da Oracle. Por fim, o sistema PostgreSQL mostrou ter ferramentas que ainda necessitam de melhorias e o incremento de tecnologias mais dinâmicas e flexíveis para auxiliar com mais eficiência o trabalho do administrado do banco de dados.

Conclui-se que o objetivo do trabalho está atingido, pois as ferramentas de auxílio à *tuning* orientaram corretamente o administrador a realizar melhorias nas consultas e na estrutura do banco de dados. Os ganhos de desempenho podem ser vistos nos resultados finais de cada consulta (salvo alguma limitação da ferramenta), a maior parte das consultas (*queries*) foi otimizada tanto em nível de tempo quanto em nível de custo de execução.

REFERÊNCIAS

BURGO, A. F. **Tuning de Banco de Dados em um software de CRM**, 2007. Disponível em: <<http://www2.dc.uel.br/nourau/document/?view=467>>. Acesso em: 01 jan. 2011.

DATE, C. J. **Introdução a Sistemas de Bancos de Dados**. Campus, São Paulo, 2005.

ELMASRI, R. A.; NAVATHE, S. **Sistema de Banco de Dados**. Person, São Paulo, 2004.

IKEMATU, R. S. **Realizando tuning na base de aplicações**, 2002. Disponível em: <<http://www.batebyte.pr.gov.br/modules/conteudo/conteudo.php?conteudo=1592>>. Acesso em: 08 maio 2011.

GORGES, C. E. **Processamento e otimização de consultas em Gerenciadores de Banco de Dados Relacionais**. 2008. Disponível em: <<http://www.geocities.ws/carlos.gorges/Monografia-CarlosGorges.pdf>>. Acesso em: 21 maio 2011.

LIFSCHITZ, S.; MILANES, A. Y.; SALLES, M. A. V. **Estado da arte em auto-sintonia de sistemas de Bancos de Dados Relacionais**. In: Technical report, Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro. PUC-Rio, 2004.

MULLINS, C. S. **Database Fundamentals**. 2010. Disponível em: <http://www.craigsmullins.com/dbta_108.html>. Acesso em: 11 abr. 2011.

PARRA, L. S. **Tuning de Desempenho em Banco de Dados**. 2005. Disponível em: <www2.dc.uel.br/nourau/document/?down=568>. Acesso em: 12 jul. 2011.

POSTGRES ENTERPRISE MANAGER GETTING STARTED GUIDE, 2007. Disponível em: <<http://www.enterprisedb.com/download-postgres-enterprise-manager>>. Acesso em: 02 nov. 2011.

SILBERSCHATZ, A.; KORTH, H.; SUDARSHAN, S. **Sistema de Banco de Dados**, 3° ed. Rio de Janeiro: Makron, 1999.

_____. **Sistema de Banco de Dados**, 5° ed., Rio de Janeiro: Elsevier, 2006.

TELLES, M. J. **Desenvolvimento de uma Ferramenta para Análise de Ajuste de Desempenho do Banco de Dados Oracle**. 2009. Disponível em: <<http://siaibib01.univali.br/pdf/Jonas%20Rosa.pdf>>. Acesso em: 12 jul. 2011.