



ISSN 1677-8464

Indexação Textual usando Sistemas Gerenciadores de Banco de Dados Livres

Isaque Vacari¹
Marcos Cezar Visoli²

Devido o crescimento de dados organizados em meios eletrônicos e a necessidade de encontrá-los rapidamente, faz-se necessário a criação de ferramentas de busca de dados e/ou informações em textos a partir de expressões de busca. Entretanto, a busca de palavras em enormes seqüências de caracteres (*strings*) pode ser extremamente lenta em termos de performance computacional, tornando a busca difícil para o usuário.

Uma das alternativas encontradas para esse problema consiste em dividir o texto a ser pesquisado em células maiores, denominadas palavras-chave, onde o mecanismo de busca irá atuar para realizar as tarefas de indexação e recuperação de dados. A utilização de palavras-chave reduz substancialmente o número de elementos distintos a serem indexados, permitindo que algoritmos mais eficientes possam ser implementados para reduzir o tempo de busca. Por exemplo, o uso da estrutura de dados dinâmica árvore, que visa o armazenamento de coleções ordenadas, pode ser aplicado para organizar todas as palavras de um contexto otimizando todo processo de busca de uma expressão.

Busca textual baseada em palavras-chave pode ser encontrada na *internet*, como por exemplo, os *sites web* de busca (Yahoo, 2005; Google, 2005; Aonde Network, 2005) etc., que procuram por palavras-chave em documentos guardados em gigantescas bases de dados.

Esse estudo está centrado na utilização de mecanismos de busca integrados a Sistemas Gerenciadores de Banco de Dados (ou simplesmente SGBD) livres e gratuitos. Mais especificamente serão abordados os SGBDs PostgreSQL - versão 8.0.1 (The PostgreSQL Global Development Group, 2005) e MySQL - versão 4.1.10a (MySQL AB, 2005), por oferecerem uma camada de indexação nativa ao banco de dados, o que não acontece com o SGBD Firebird (Firebird Project, 2005) e demais SGBDs livres, que precisam usar outros programas, geralmente proprietários, para realizar as atividades de indexação.

Existem outras opções de indexadores independentes de SGBD que atuam sobre documentos estáticos (documentos HTML, documentos de texto, PDF etc.), por exemplo: SWISH-E – Simple Web Indexing System for Humans - Enhanced (Swish-e, 2005), LUCENE (The Apache Software Foundation, 2005b), OpenFTS - FULL TEXT Search Engine (XWare, 2005) e o próprio indexador do Google (Google, 2005) que indexa automaticamente páginas disponíveis na *internet*. A lista completa desses indexadores pode ser consultada em Search Tools Consulting (2005), e maiores detalhes foram estudados no projeto de implementação do serviço Agência de Informação Embrapa (Cruz, 2003).

Para aplicações, cujos dados são armazenados em banco de dados (sob gerência de um SGBD) e que precisam utilizar

¹ Tecnólogo em Processamento de Dados, Técnico de Nível Superior da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP. (e-mail: isaque@cnptia.embrapa.br)

² Bacharel em Ciência da Computação, Pesquisador da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP. (e-mail: visoli@cnptia.embrapa.br)

ferramentas de indexação baseadas em documentos estáticos para geração e busca de dados, a alternativa é criar uma camada intermediária entre a aplicação e o banco de dados. Essa camada é composta pelos registros do banco de dados em forma de palavras-chave, gerando um banco de índices, sob a qual deverá ser construída a aplicação para busca e apresentação dos dados.

Essa solução é extremamente custosa em termos de desenvolvimento de *software*, pois além de gerar duplicidade de dados no banco de índices, é necessário escrever um programa para sincronizar os dados armazenados no banco de dados com aqueles guardados no banco de índices, de forma que, todo registro inserido no banco de dados tenha um elemento vinculado no banco de índices. A situação ideal é automatizar esse trabalho, deixando, também, por conta do SGBD as tarefas relativas à indexação.

Assim, o uso de SGBDs com indexador nativo ao banco de dados, torna-se uma alternativa eficaz para projetos que necessitem utilizar um mecanismo de busca avançado. Além disso, oferece as seguintes vantagens:

- dispensa a camada de *software* para criação e manutenção de índices, deixando essa atividade transparente para a aplicação e o desenvolvedor;
- independência da linguagem de programação para desenvolvimento da aplicação de busca e apresentação dos dados, pois o elemento indexador está vinculado diretamente ao SGBD. Nesse perfil se enquadra os SGBDs PostgreSQL e MySQL³, que além de serem livres, gratuitos e operarem em diversos sistemas operacionais, oferecem alternativas de indexação nativa ao banco de dados.

Principais Recursos oferecidos por um Mecanismo de Busca

Os principais recursos de busca podem ser encontrados em diversos *sites web*, desde os mais conhecidos como Google (Google, 2005), até aqueles de uso mais específico, como o Compendex (Elsevier Engineering Information, 2005) e a BDPA (Embrapa Informática Agropecuária, 2005a). Os recursos de busca geralmente permitem efetuar desde consultas simples (com resultados mais abrangentes) até consultas mais sofisticadas (com resultados mais específicos). Dentre a variedade de opções de busca, as mais significativas são:

Operadores booleanos: permitem a combinação de termos de busca, podendo ampliar ou reduzir o escopo da pesquisa. Os operadores booleanos são: **e**, **ou** e **não**, correspondentes aos conhecidos termos em inglês **and**, **or** e **not**:

- **E** (interseção): restringe a busca, recuperando apenas os registros que contêm todos os termos da expressão de busca. Exemplo: **arroz e feijão** (recupera referências sobre arroz e feijão);
- **ou** (união): amplia a busca, recuperando todos os registros que contêm qualquer um dos termos da expressão de busca. Exemplo: **milho ou soja ou trigo**

(Recupera referências que contêm qualquer uma dessas palavras);

- **não** (negação): deve ser utilizado quando se deseja excluir um determinado conjunto de dados. Exemplo: **milho e não semente** (recupera os registros que contêm a palavra milho mas não contêm a palavra semente).

Truncagem: o asterisco (*) colocado à direita do termo, tem a função de recuperar todos os termos com o prefixo ou radical dado. Exemplo: a expressão **brasil*** recupera **brasil**, **brasileiro**, **brasileira**, **brasileiros**, **brasileiras**, **Brasilia**, **brasiliense**, **brasilis**, etc.

Mascaramento: o mascaramento (representado pelo caracter ?) tem a função de substituir qualquer caracter em um termo de busca. Por exemplo, a expressão: **bra?il** recupera **Brasil** e **Brazil**.

Busca por proximidade: o comando **proximo** (*near*) recupera termos adjacentes, ou seja, que aparecem no mesmo campo do registro, mas próximo um do outro. Exemplo: **lagarta proximo soja** recupera lagarta da soja, mas não recupera registros em que a palavra lagarta esteja distante da palavra soja.

Busca por frase: é possível recuperar frases e expressões, colocando o termo ou a expressão de buscas, geralmente, entre aspas simples (') ou duplas ("). Exemplo: '**Cruz das Almas**' ou "**cruz das almas**". Com este recurso, também é possível recuperar frases ou expressões que contêm . (ponto), : (dois pontos) e ; (ponto e vírgula), como, por exemplo, **endereços eletrônicos**. Exemplo: '**linux@softwarelivre.org**'.

Busca por campo específico: é possível implementar consultas sobre um ou mais campos especificados. Exemplo: **software em Titulo**, recupera todas as ocorrências que possuem a palavra *software* no campo Titulo.

Agrupamento: permite associar várias expressões através da utilização de parênteses, exemplo: (arroz ou feijão) e soja. Retorna arroz e soja, ou então, feijão e soja.

Palavras de parada (*stopwords*): a utilização de palavras de parada permite não incluir no mapa de índices palavras indesejadas, como: artigos, preposições etc.

Tipos de campo: suporte à indexação de campos do tipo alfanumérico, numérico e datas.

Ordenação por relevância: relevância significa ordenar o resultado da consulta por registros onde ocorreram maior incidência do termo pesquisado.

Suporte à pesquisa acentuada: trata da mesma maneira palavras acentuadas e não acentuadas. Exemplo: **programacao**, deve retornar **programacao** (sem acento) e **programação** (com acento).

Busca com tratamento de ênfase: significa buscar duas ou mais palavras definindo prioridade. Exemplo: **menor arroz maior feijão**, retorna todos os registros onde a palavra arroz possui um número menor de ocorrências que a palavra feijão.

Destaque: inclusão de marca automática sobre o texto encontrado.

³ Para distribuição em aplicações comerciais o MySQL prevê a cobrança de *royalties*.

Existem outros recursos de busca, como por exemplo, o *stemm* (Snowball, 2005) implementado pelo *site web* de busca Compendex (Elsevier Engineering Information, 2005), que não foram analisados nesse estudo.

Mecanismos de Busca e Indexação em SGBDs Livres

Tsearch2

O Tsearch2 (Generalized Search Tree, 2005b) é a segunda geração de ferramentas de indexação de texto criado para o PostgreSQL, desenvolvido por Oleg Bartunov e Teodor Sigaev e faz parte do PostgreSQL desde a versão 7.4. É formado por uma coleção de operadores, funções e recursos adicionais que permitem ao PostgreSQL criar, manter e recuperar registros indexados pelo mecanismo de palavras-chave.

O Tsearch2 parte do princípio que é possível dividir um texto em átomos maiores (palavras-chave), e dispor essas unidades em vetores que serão indexados em árvores de pesquisas genéricas denominadas GiST (Generalized Search Tree, 2005a). Os índices de palavras-chave não são criados diretamente sobre o campo com o conteúdo dos dados (textos), sendo necessário criar um campo adicional, na própria tabela de dados com estrutura própria para indexação. Esse campo adicional deverá ser sincronizado com o campo de dados original, visando garantir que todas as modificações ocorridas no campo original sejam refletidas no campo adicional indexado. A Fig. 1 ilustra o funcionamento da arquitetura Tsearch2.

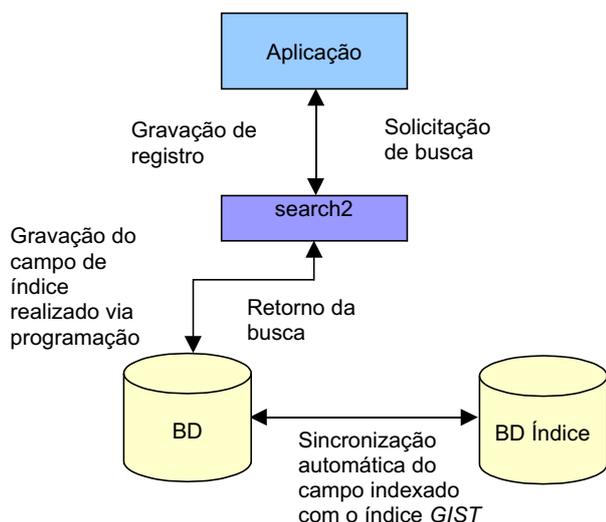


Fig. 1. Arquitetura Tsearch2 para PostgreSQL.

FULL TEXT

O MySQL utiliza o índice especial FULL TEXT para realizar as operações de indexação. Esse índice está presente desde a versão 3.23.23 do MySQL e atua diretamente sobre o conteúdo do campo de dados. Essa implementação torna transparente e automático o processo de atualização e sincronização do campo de dados com o campo de índice. A Fig. 2 ilustra o funcionamento da arquitetura Full Text.

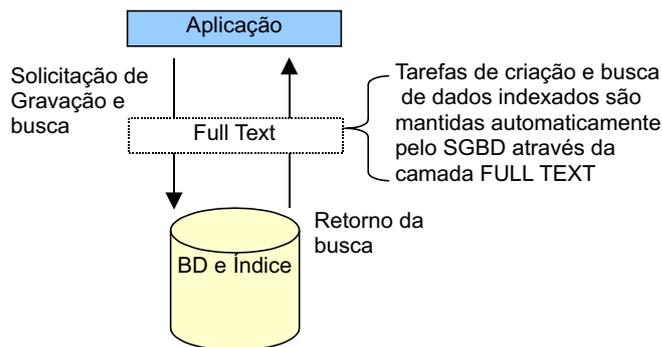


Fig. 2. Arquitetura FULL TEXT para MySQL.

Estudo de Caso – Base de Dados da Pesquisa Agropecuária

Para comprovar o desempenho dos mecanismos citados, foi escolhido a Base de Dados da Pesquisa Agropecuária (Embrapa Informática Agropecuária, 2005a) com ênfase para o Acervo Documental da Embrapa (Embrapa Informática Agropecuária, 2005b). A coleção Base de Dados da Pesquisa Agropecuária (ou simplesmente BDPA) torna viável o acesso à literatura técnico-científica produzida e colecionada pelas Unidades da Embrapa em todo território nacional. Essa coleção é formada pelas seguintes bases de dados: Acervo Documental Embrapa, Produção Científica da Embrapa, Catálogo Coletivo de Periódicos da Embrapa e Cadastro de Instituições.

A tabela de dados **Acervo Documental Embrapa**, é composta por aproximadamente 400.000 registros. A arquitetura para consulta ao ACERVO e demais bases de dados foi desenvolvida utilizando o SGBD livre e gratuito Firebird (Firebird Project, 2005), a linguagem de programação proprietária Delphi (Borland Software Corporation, 2005), e o indexador proprietário Rubicon (Tamarack Associates, 2005), todos obrigatoriamente atuando sobre o sistema operacional Windows (marca registrada de Microsoft Inc.).

A utilização do indexador RUBICON tem sido a alternativa escolhida para implementação de busca textual para o SGBD Firebird, devido ao motivo do Firebird não possuir indexador nativo para realização das tarefas de indexação.

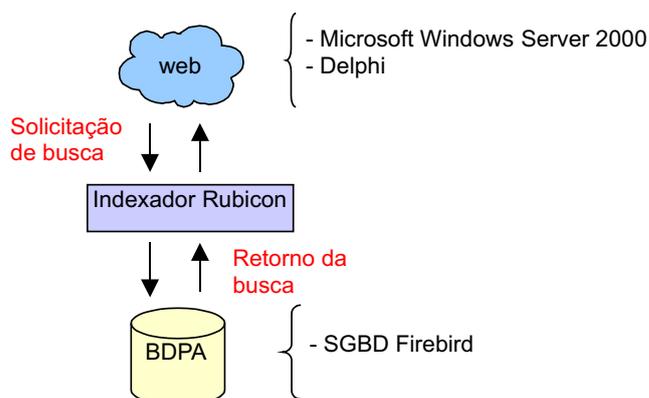


Fig. 3. Arquitetura atual da BDPA.

Baseado nessa arquitetura decidiu-se encontrar e aplicar soluções baseadas em *software* livre (em termos de SGBD e sistema operacional) como alternativas sustentáveis para o modelo atual implementado para BDPA. Inicialmente, foi explorado os SGBDs PostgreSQL com o indexador nativo Tsearch2 e o MySQL com o indexador nativo FULL TEXT. Os próximos passos consistem em demonstrar os resultados obtidos em termos de recursos oferecidos e desempenho pelos indexadores Tsearch2, FULL TEXT e Rubicon.

Comparação entre os Recursos oferecidos pelos Mecanismos de Busca

A Tabela 1 apresenta a comparação dos mecanismos de busca baseados em SGBDs Livres (PostgreSQL e MySQL) e da arquitetura proprietária atual (RUBICON) em relação aos principais recursos que devem fazer parte de uma poderosa camada de indexação.

Tabela 1. Recursos oferecidos pelos mecanismos de busca.

Recurso	Firebird + Rubicon	PostgreSQL + Tsearch2	MySQL + FULL TEXT
Operadores Booleanos	Sim	Sim	Sim
Truncagem	Sim	Não – mas, pode ser resolvido usando o algoritmo de redução de palavras ao radical para português, espanhol e inglês	Sim
Mascaramento	Sim	Não	Sim – resolvido com truncagem
Busca por proximidade	Sim	Não	Não
Busca por frase	Sim	Não	Sim
Busca por campo específico	Sim	Sim – criando um campo indexado para cada campo de dado	Sim
Agrupamento	Sim	Sim	Sim
Palavras de parada	Sim	Sim	Sim
Busca com tratamento de ênfase	Não	Não	Sim
Tipos de campo	Alfanuméricos, numéricos e datas	Alfanuméricos, numéricos e datas	Somente alfanuméricos
Relevância	Sim	Sim	Sim
Pesquisa acentuada	Sim, através de programação específica na aplicação	Não	Sim
Destaque	Não	Sim	Não

Comparação de Desempenho

Infra-estrutura de *hardware e software*

Para uma análise mais detalhada, em termos de tempo de resposta a partir de uma busca, foi implementada a tabela de dados ACERVO (composta por aproximadamente 400.000 registros e 25 campos, dos quais 22 campos são indexados) nos SGBDs MySQL e PostgreSQL, para isso utiliza-se as seguintes ferramentas:

- GNU/Linux SUSE 9.4 com 256MB de memória RAM e processador Intel(R) Pentium(R) 4 CPU 2.80 GHz;
- Linguagem de programação Java (Sun Microsystems, 2005) e interface de programação Java NetBeans 4.1 (Netbeans Org., 2005) para criação da aplicação para migração dos dados da tabela de ACERVO para os SGBDs PostgreSQL e MySQL;
- DBVisualizer (Minq Software, 2005) como ferramenta para análise e modelagem do banco de dados;

- Apache 2.0 Web Server (The Apache Software Foundation, 2005a), utilizado como servidor *Web* visando fornecer serviços para implementação de consulta(s) aos SGBDs MySQL e PostgreSQL em uma interface *web*;
- phpPgAdmin (PhpPgAdmin Development Team, 2005), interface *web* para administração do PostgreSQL e execução de comandos SQL – *Structure Query Language*
- PgAdmin III (PgAdmin Development Team, 2005), interface *web* para administração do PostgreSQL mais oferecida em forma de aplicação *desktop*;
- PgMyAdmin (PhpMyAdmin Development Team, 2005), interface *web* para administração do MySQL e execução de comandos SQL – *Structure Query Language*.

Conjunto de consultas, testes e resultados

Como exemplo, foi elaborado um pequeno conjunto de consultas para verificar o desempenho dos métodos utilizados. Foram escolhidas expressões que atuam sobre operadores booleanos, frases, em campos específicos e truncagem. A execução dos testes foi realizada com as ferramentas PgAdmin III e phpMyAdmin que retornam o tempo de resposta e os registros encontrados em uma operação de pesquisa.

Também foi replicada todas as consultas efetuadas na BDPA (que utiliza o indexador RUBICON), com o propósito de avaliar o desempenho dos indexadores baseados em SGBDs livres. Vale ressaltar que a BDPA está implementada em um computador distinto (com configurações de *hardware e software* diferentes da infra-estrutura implementada para os testes com os indexadores Tsearch2 e FULL TEXT).

Tabela 2. Testes efetuados com tempo de resposta (em segundos) dos mecanismos de busca.

Expressão de busca	PostgreSQL + Tsearch2	MySQL + FULL TEXT	Firebird + RUBICON (BDPA)
procurar por java em todos os campos	840 segundos	0,4032 segundos	0,109 segundos
procurar por java e prt em todos os campos	651 segundos	0,7315 segundos	0,156 segundos
procurar por java nos campos título e palavras-chave	17,109 segundos	5,7563 segundos	Implementa, mas não disponível
procurar por java e linguagem nos campos título e palavras-chave	17,094 segundos	9,2345 segundos	Implementa, mas não disponível
procurar por java contendo a frase "Linguagem de Programação" em todos os campos	não implementa busca por frase	0,5086 segundos	16,219 segundos
procurar por bras* e agri* em todos os campos	não implementa truncagem	2,2345 segundos	3,407 segundos

Ponderações

Durante a implementação dos mecanismos de busca textual pertencentes aos SGBDs PostgreSQL e MySQL e RUBICON, foram observados algumas ponderações importantes presentes nos três indexadores, relatados na Tabela 3.

Tabela 3. Ponderações dos mecanismos de busca.

Ponderações	PostgreSQL + Tsearch2	MySQL + FULL TEXT	Firebird + RUBICON (BDPA)
Performance	Razoável se houver uma redução drástica no número de campos a serem indexados, como também, no número de campos a serem tratados em uma pesquisa – depende de ajuste fino (otimização de memória, tratamento de <i>cache</i> de consultas etc.) no PostgreSQL.	Altamente satisfatória.	Altamente satisfatória.
Restrições de tabela	Campo de índice é armazenado diretamente na tabela que contem os dados.	Número máximo de campos indexados por tabela: 16.	Não há.
	Tamanho máximo do campo de índice por registro: 1MB. Desempenho está associado a um número reduzido de palavras por índice.	Aplicado somente em tabelas do tipo MyISAM (não suportam integridade referencial, <i>store procedures</i> , <i>triggers</i> etc.).	
Licença	Gratuito.	Gratuito, mas para distribuição em aplicações comerciais é necessário o pagamento de <i>royalties</i> .	Proprietário, o que eleva o custo da aplicação e distribuição da BDPA para novos interessados. Como também, exige o pagamento de licença em melhorias e correções de erros implementadas pelo componente indexador RUBICON.

Considerações Finais

A possibilidade de integrar o mecanismo de busca à base de dados relacionais tem ganhado grande aceitação para construção de diversos tipos de ferramentas de busca, como: *sites*, sistemas de catalogação eletrônica, armazém de dados etc.

Atualmente o PostgreSQL e o MySQL são os SGBDs livres que mais se destacam no assunto indexação textual, no entanto, ambos estão em processo de evolução. Dentro do estudo realizado, o PostgreSQL é indicado para aplicações mais "robustas" e complexas que necessitam de transações, suporte à integridade referencial etc., mas com um conjunto menor de campos a serem indexados.

Já o MySQL é mais indicado para construção de aplicações de consulta, geralmente disponibilizadas em uma interface *web*, que efetuam pesquisas em enormes base de dados, com um número maior de campos indexados, com tempo de resposta reduzido devido a complexidade da aplicação ser menor.

Referências Bibliográficas

AONDE NETWORK. **Aonde.com.br – seu site de busca brasileiro!** Disponível em: <<http://www.aonde.com.br>>. Acesso em: 07 nov. 2005.

THE APACHE SOFTWARE FOUNDATION. **The Apache Software Foundation [home page]**. Disponível em: <<http://www.apache.org>>. Acesso em: 04 ago. 2005.

THE APACHE SOFTWARE FOUNDATION. **Lucene**. Disponível em: <<http://lucene.apache.org>>. Acesso em: 4 ago. 2005.

BORLAND SOFTWARE CORPORATION. **Borland IDE: Delphi – Windows .NET application development tool w/ object relational mapping**. Disponível em: <<http://www.borland.com/delphi>>. Acesso em: 17 nov. 2005.

CRUZ, S. A. B. da. **Implantação de um serviço de busca em site da WWW**. Campinas: Embrapa Informática Agropecuária, 2003. 8 p. (Embrapa Informática Agropecuária. Comunicado Técnico, 50). Disponível em: <<http://www.cnptia.embrapa.br/modules/tinycontent3/content/2003/comtec50.pdf>>. Acesso em: 8 nov. 2005.

ELSEVIER ENGINEERING INFORMATION INC. **Engineering Village 2 – quick search**. Disponível em: <<http://www.engineeringvillage2.org>>. Acesso em: 8 nov. 2005.

EMBRAPA INFORMÁTICA AGROPECUÁRIA. **Bases de dados da pesquisa agropecuária**. Disponível em: <<http://bdpa.cnptia.embrapa.br>>. Acesso em: 04 ago. 2005a.

EMBRAPA INFORMÁTICA AGROPECUÁRIA. **BDPA – pesquisar: acervo documental Embrapa**. Disponível em: <http://bdpa.cnptia.embrapa.br/fr_acervo.html>. Acesso em: 04 ago. 2005b.

FIREBIRD PROJECT. **Firebird - relational database for the new millenium**. Disponível em: <<http://firebird.sourceforge.net>>. Acesso em: 05 ago. 2005.

GENERALIZED SEARCH TREE. **The GiST: indexing project**. Disponível em: <<http://gist.cs.berkeley.edu>>. Acesso em: 6 ago. 2005.

GENERALIZED SEARCH TREE. **Tsearch2 - full text extension for PostgreSQL**. Disponível em: <<http://www.sai.msu.su/~megera/postgres/gist/tsearch/V2>>. Acesso em: 6 ago. 2005.

GOOGLE. **Google Brasil**. Disponível em: <<http://www.google.com.br>>. Acesso em: 7 nov. 2005.

MINQ SOFTWARE AB. **Minq Software – products: DbVisualizer**. Disponível em: <<http://www.dbvis.com>>. Acesso em: 19 set. 2005.

MINQ SOFTWARE. **DbVisualizer**. Disponível em: <<http://www.dbvis.com/products/dbvis/>>. Acesso em 6 ago. 2005.

MYSQL AB. **MySQL: the world's most popular open source database**. Disponível em: <<http://www.mysql.com/>>. Acesso em: 5 ago. 2005.

NETBEANS. ORG. **Netbeans [home page]**. Disponível em: <<http://www.netbeans.org>>. Acesso em: 19 set. 2005.

PGADMIN DEVELOPMENT TEAM. **pgAdmin III: PostgreSQL administration and management tools**. Disponível em: <<http://www.pgadmin.org>>. Acesso em: 19 set. 2005.

PHPMYADMIN DEVELOPMENT TEAM. **phpMyAdmin**: MySQL database administration tool: the phpMyAdmin project. Disponível em: <<http://www.phpmyadmin.net>>. Acesso em: 5 out. 2005.

PHPPGADMIN DEVELOPMENT TEAM. **phpPgAdmin**: web based PostgreSQL administration tool. Disponível em: <<http://phppgadmin.sourceforge.net>>. Acesso em: 19 set. 2005.

THE POSTGRESQL GLOBAL DEVELOPMENT GROUP. **PostgreSQL**: the world's most advanced open source database. Disponível em: <<http://www.postgresql.org/>>. Acesso em: 5 ago. 2005.

SEARCH TOOLS CONSULTING. **Alphabetical list of SearchTools product reports**. Disponível em: <<http://www.searchtools.com/tools/tools.html>>. Acesso em: 26 out. 2005.

SNOWBALL. Disponível em: <<http://snowball.tartarus.org>>. Acesso em: 17 nov. 2005.

SUN MICROSYSTEMS, INC. **Java technology**. Disponível em: <<http://java.sun.com>>. Acesso em: 19 set. 2005.

SWISH-E. **Simple Web Indexing System for Humans – Enhanced**. Disponível em: <<http://swish-e.org>>. Acesso em: 4 ago. 2005.

TAMARACK ASSOCIATES. **[Rubicon]**. Disponível em: <<http://www.tamaracka.com>>. Acesso em: 7 ago. 2005.

XWARE TEAM. **OpenFTS**: open source full text search engine. Disponível em: <<http://openfts.sourceforge.net>>. Acesso em: 4 ago. 2005.

YAHOO. **Yahoo! Brasil**. Disponível em: <<http://br.yahoo.com>>. Acesso em: 7 nov. 2005.

Comunicado Técnico, 70

Ministério da Agricultura, Pecuária e Abastecimento



Embrapa Informática Agropecuária
Área de Comunicação e Negócios (ACN)
Endereço: Caixa Postal 6041 - Barão Geraldo
13083-970 - Campinas, SP
Fone: (19) 3789-5743
Fax: (19) 3289-9594
e-mail: sac@cnptia.embrapa.com.br

1ª edição on-line - 2005

© Todos os direitos reservados.

Comitê de Publicações

Presidente: Kleber Xavier Sampaio de Souza.
Membros Efetivos: Adriana Farah Gonzalez (secretária), Ivanilde Dispatto, Luciana Alvim Santos Romani, Marcia Izabel Fugisawa Souza, Renato Fileto, Stanley Robson de Medeiros Oliveira.

Suplentes: José Iguelmar Miranda, Laurimar Gonçalves Vendrusculo, Maria Goretti Gurgel Praxedis, Sílvio Roberto Medeiros Evangelista.

Expediente

Supervisor editorial: Ivanilde Dispatto
Normalização bibliográfica: Marcia Izabel Fugisawa Souza
Editoração eletrônica: Área de Comunicação e Negócios