



Confira
tudo que
rolou no
maior
evento
mundial
sobre
Firebird!

2º
www.**Firebird**
Developers
Day.com.br

Especial

Editorial

Acho que todos perceberam já pela capa, que essa se trata de uma edição no mínimo “diferente” da DB FreeMagazine! O principal motivo é que nessa edição estaremos tratando exclusivamente do **maior evento de Firebird do mundo**, o **Firebird Developers Day (FDD)**, que aconteceu no dia **16 de Julho de 2005** em Piracicaba, SP.

Essa foi a segunda edição do evento, que é realizado e organizado anualmente pela FireBase, ou mais especificamente, por mim :-)

Nesse ano tivemos um brilho extra no evento, devido a presença de dois ícones internacionais da comunidade do Firebird: **Jim Starkey** (criador do InterBase e do Firebird Vulcan) e **Ann Harrison** (chefe do depto. técnico da IBPhoenix). Além disso, batemos o recorde de inscritos: mais de **650 pessoas!** Esse número fez com que o FDD seja hoje o maior evento de banco de dados Firebird do mundo (referindo-se ao número de participantes).

No período da manhã, tivemos **seis palestras** sobre temas variados, acontecendo sempre ao pares, em paralelo. Após o almoço, tivemos mais **4 palestras** sem concorrência.

A estrutura utilizada no evento é uma das melhores do Brasil, e constituiu-se de um **teatro** com capacidade para **800 pessoas**, e uma **sala para 200 pessoas**. Aproveito para agradecer a UNIMEP, e mais especificamente ao pessoal do curso de Sistemas de Informação, por mais uma vez ter nos disponibilizado essa estrutura maravilhosa!

*Estaremos lançando em breve um **DVD** com as palestras apresentadas, assim todos que não puderam comparecer no evento poderão se beneficiar do conhecimento que foi transmitido lá. **O lucro das vendas do DVD será remetido para a Fundação Firebird!***

Um report completo com fotos está disponível no site do evento: www.FirebirdDevelopersDay.com.br

Gostaria de agradecer a todos os palestrantes, patrocinadores e pessoas envolvidas, bem como todos que estiveram presentes no evento! Espero vê-los novamente em 2006, na terceira edição do FDD!

Up the Irons!

Carlos H. Cantu
Editor DB FreeMagazine

ANUNCIE NA DB FreeMagazine

Valorize seu produto ou serviço!

anuncios@dbfreemagazine.com.br

Informações

DB FreeMagazine nº 004 - Ano I
Julho/2005
Contato geral:
webmaster@dbfreemagazine.com.br

Equipe editorial

Carlos H. Cantu
(cantu@dbfreemagazine.com.br)

Luiz Paulo de Oliveira Santos
(lpaulo@dbfreemagazine.com.br)

Contribuíram nessa edição

Luiz Paulo de Oliveira Santos
Carlos Henrique Cantu
Eduardo Mercado
Emerson Facunte
Eduardo Rocha
Ann Harrison
Jim Starkey
José Alberto Rodrigues Filho (Beto)
Julio, Gustavo e Max (Remédio em Casa)

É proibida a reprodução de qualquer parte do conteúdo dessa publicação sem autorização prévia por escrito.

Dica para melhor visualização:

Utilize a resolução 1024x768 pixels e configure o *Acrobat Reader* para Zoom de 100%. Feche todas as abas laterais e esconda as barras de ferramentas, liberando o máximo de área útil na tela, ou simplesmente rode a revista em modo *fullscreen*.

Saindo do Forno...

Upgrades para o Oracle

A Oracle informa que pretende lançar em breve uma grande atualização para seus produtos de BI e OLAP. As novidades vão desde a correção de alguns problemas, até a implementação de novos recursos, introdução de novos wizards, remodelagem da interface, etc.

Fonte: <http://www.rittman.net/archives/001318.html>

Microsoft planeja mudanças nas certificações

O lançamento do MSSQL Server em Novembro de 2005 deve trazer também mudanças no esquema de certificações atuais oferecidas pela Microsoft. A gigante planeja remodelar alguns tópicos, focando mais especificamente o SQL Server e o Visual Studio 2005.

Fonte: <http://www.computerworld.com/databasetopics/data/software/story/0,10801,103355,00.html>

Firebird 2.0 alpha 3 é lançado

Uma nova versão de avaliação do Firebird

DBManager Professional

Gerenciamento para seus Bancos de Dados

- Suporte à Múltiplos SGBD's
- Diagram Designer
- Gerador de Formulários e Relatórios
- Gerenciador de Tarefas
- Controle de Versões do Banco de Dados
- Editor, Debugger e Planner para Consultas
- Monitore Servidores, Bancos de Dados e Tabelas
- E muito mais.

DBTools Software
www.dbtools.com.br

2.0 acaba de ser lançado. Entre as novidades incluídas nessa versão estão os novos collates para o português do Brasil, possibilitando a ordenação e comparação de strings de forma a não diferenciar maiúsculas de minúsculas, e caracteres acentuados de não acentuados. Pela primeira vez, uma versão do Firebird está trazendo mudanças que tiveram a participação direta de brasileiros no seu desenvolvimento!

Fonte: <http://www.FireBase.com.br>

MySQL anuncia melhor trimestre de todos os tempos

A MySQL acaba de anunciar que o trimestre de Abril/Junho de 2005 foi o mais lucrativo

Nova versão 3.1.0

Enterprise Edition

Suporte à Oracle, MSSQL Server, Sybase, MSAccess, ODBC, MySQL, PostgreSQL, Firebird, Interbase, XBase e SQLite.

Download gratuito da
Freeware Edition
Inclui MySQL, PostgreSQL, Firebird, Interbase, XBase e SQLITE

de toda a sua história. Ela acredita que boa parte disso se deve ao novo serviço comercial chamado "MySQL Network", que compreende um pacote de softwares certificados, suporte especial, etc. O lucro do trimestre foi mais que o dobro do mesmo período do ano passado.

Fonte: http://www.mysql.com/news-and-events/press-release/release_2005_26.html

Brasil tem a maior conferência internacional de Firebird

Com mais de 650 inscritos, o Firebird Developers Day se tornou o maior evento de banco de dados Firebird de todo o mundo. Realizado anualmente em Piracicaba-SP, o evento vem conquistando cada vez mais participantes, e



Seja um AUTOR da Editora Ciência Moderna!

Se você possui algum manuscrito, original ou livro pronto, submeta à nossa apreciação enviando um resumo do seu trabalho para o e-mail: lcm@lcm.com.br. Teremos o maior prazer em avaliá-lo e, se aprovado, publicá-lo em forma de livro.

mostra que o Brasil provavelmente tem hoje a maior base de usuários de Firebird no mundo.

Fonte: <http://www.FirebirdDevelopersDay.com.br>

Empresas adotam Open Source porque é melhor, e não só porque é mais barato

Um artigo disponibilizado no portal britânico TechWorld afirma que as empresas europeias têm adotado softwares Open Source porque são confiáveis e mais flexíveis, e não somente porque são mais baratos. Esse é o resultado de uma pesquisa feita pela IDC.

Fonte: <http://www.techworld.com/opsys/news/index.cfm?newsid=3535>

Teste de qualidade com SGBDR baseado no PostgreSQL

A empresa EnterpriseDB Corporation, que utiliza um banco de dados baseado no PostgreSQL, anunciou que seu SGBDR passou no teste de qualidade da Coverity Prevent, a mais avançada ferramenta de análise de software do mundo, que encontrou apenas 20 problemas em mais de 775.000 linhas de código. Os problemas já foram corrigidos e serão aplicados nas próximas versões oficiais

do PostgreSQL e do EnterpriseDB 2005, provavelmente no final do verão americano.

Fonte: <http://www.postgresql.org/about/news.363>

Artigo discute Bancos de Dados Open Source e dá dicas para migração

Um artigo publicado pela Virtuas Solutions fala sobre a adoção de bancos de dados Open Source, critérios para decisão sobre qual software adotar, suporte, ferramentas para migração, etc. A versão em inglês do artigo pode ser vista em <http://virtuas.com/osl-osrdb-01.pdf>



Firebird Essencial

Primeiro livro brasileiro que trata especificamente dos recursos do SGBD Firebird (versões 1.0 e 1.5). O autor reuniu no livro todo o material produzido por ele para as revistas ClubeDelphi e SQLMagazine. Os artigos foram revisados, atualizados e muitos deles complementados, de forma a proporcionar ao leitor uma fonte de informação rica, atualizada e confiável. Um capítulo inédito sobre a criação de UDFs foi escrito exclusivamente para o livro.

Você aprenderá a instalar o SGBD, criar procedures, catálogos em CDROM, criar backups, gerenciar usuários, utilizar campos BLOB de forma adequada, identificar os tipos de dados disponíveis no Firebird, e muito mais!

Verifique o sumário do livro em www.firebase.com.br/fb/livro/fbessencial

www.firebase.com.br

Apresentação

ClientDataSet

Instrutor

Eduardo Rocha

EduDelphiPage
<http://www.edudelphipage.com.br>

Revista Active Delphi
<http://www.activedelphi.com.br>

Resumo

A tecnologia dbExpress juntamente com o ClientDataSet é hoje uma das principais formas de acesso ao Firebird em aplicações desenvolvidas com as ferramentas da Borland. A palestra trará dicas de utilização do ClientDataSet, utilização de generators com dbExpress, trabalhando com mestre/detalhe, cuidados a serem tomados, drivers de terceiros para Firebird, controle transacional, etc.

Palestrante

Eduardo Rocha é desenvolvedor Delphi há 7 anos, possuindo sua própria consultoria desde 2003. É Coordenador Editorial da revista Active Delphi, mantenedor do site EduDelphiPage e um dos coordenadores do DUG-BR. Atualmente está cursando Sistemas de Informação na Universidade FIAP.

ClientDataSet - Introdução

- Utilização
- Complexidade
- Custo
- Requisitos

EduDelphiPage
<http://www.edudelphipage.com.br>

Revista Active Delphi
<http://www.activedelphi.com.br>

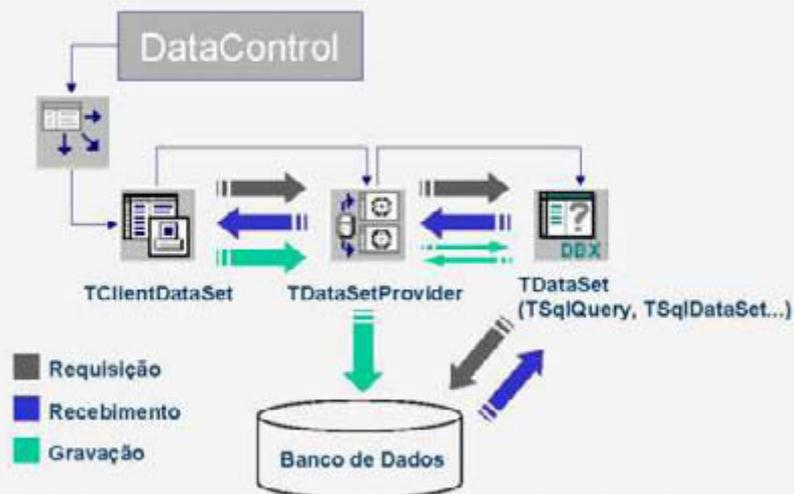
ClientDataSet - Recursos/Vantagens

- Compatibilidade com qualquer Banco de Dados
- Redução no tempo de transações abertas
- Tabelas em memória
- Gravação dos dados em outros formatos
- Busca e Indexação sem necessidade de índices físicos
- Novos tipos de campos: InternalCalc, Aggregate
- Diversos métodos para desfazer alterações - SavePoint
- Aplicações Multi-camadas, WebServices, XML, etc.

EduDelphiPage
<http://www.edudelphipage.com.br>

Revista Active Delphi
<http://www.activedelphi.com.br>

ClientDataSet - Funcionamento



EduDelphiPage
<http://www.edudelphipage.com.br>

Revista Active Delphi
<http://www.activedelphi.com.br>

ClientDataSet – Exemplos

- Exemplo 1 – Conceitos Básicos
- Exemplo 2 – Pesquisando e Filtrando Registros
- Exemplo 3 – Trabalhando com Mestre/Detalhe
- Exemplo 4 – Trabalhando com Joins
- Exemplo 5 – Trabalhando com Transações

EduDelphiPage
<http://www.edudelphipage.com.br>

Revista Active Delphi
<http://www.activedelphi.com.br>

ClientDataSet – Encerramento

Eduardo Rocha

eduardo@edudelphipage.com.br

EduDelphiPage - <http://www.edudelphipage.com.br>

Revista Active Delphi - <http://www.activedelphi.com.br>

EduDelphiPage
<http://www.edudelphipage.com.br>

Revista Active Delphi
<http://www.activedelphi.com.br>

Remédio em Casa

O Projeto Remédio em Casa é uma iniciativa pioneira da Secretaria Municipal de Saúde do Rio de Janeiro na área da assistência farmacêutica aos portadores de doenças crônicas (hipertensão arterial e ao diabetes).

- Abrangência
- Condições dos Pacientes
- Condições do Tratamento



Palestrantes:

Julio Cezar Cardoso da Silveira (juliogau@rio.rj.gov.br) é Analista de Sistemas. Possui 28 anos de experiência como desenvolvedor / gerente de projetos. Responsável técnico pelo projeto Remédio em Casa da SMS-RJ.

Gustavo Boletta (boletta@rio.rj.gov.br) é Engenheiro de Software. Possui 6 anos de experiência em desenvolvimento de sistemas OO. Trabalha na SMS-RJ há 5 anos como analista responsável pela pesquisa e desenvolvimento utilizando software livre.

Max Moura Wolosker (mwolosker@rio.rj.gov.br) é Analista de Sistemas. Possui 9 anos de experiência administrando e implantando sistemas baseados em Linux. Trabalha na SMS-RJ há 5 anos como administrador da rede.

Resumo

O Projeto Remédio em Casa é responsável pela dispensação de medicamentos para os pacientes de doenças crônicas (hipertensão e diabetes) considerados estáveis dentro dos protocolos adotados pela Secretaria Municipal de Saúde do Rio de Janeiro. São 140 servidores utilizando Firebird + Hibernate + Java com Swing nas condições mais adversas e imagináveis. Na Central, temos um servidor Firebird em condições ideais que centraliza os dados das 140 unidades de saúde, acessado por um Web Service que recebe os dados das unidades, e um aplicativo Web acessado pela Central de Distribuição.

Projeto

- Estrutura
- Funções
- Impacto Social



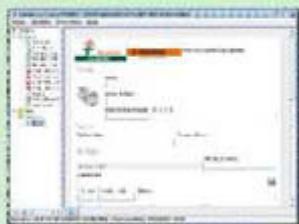
Como chegamos ao FireBird

- Remédio em Casa Versão 1
- Interbase no Delphi
- Conectividade sem a BDE
- Por quê o fork do Interbase



Porque continuar com o Firebird

- A barreira de 1GB de dados
- Interclient, JDBC, Jaybird
- Hibernate
- Sincronismo



Infra-Estrutura

- Unidades
 - Histórico das Versões
 - Hardware
 - Sistema Operacional
 - Condições
 - Acesso discado
- Nível Central
 - Histórico das Versões
 - Hardware
 - Sistema Operacional
 - Condições



Administração

- Backup
- Corrupção das Bases



Problemas

- Forced Write
- Charset e Collate
- Superserver e Multiplas conexões com drivers diversos
- Bases mal projetadas
- Consultas mal escritas



Quem Somos

Julio Cezar Cardoso da Silveira (juliogau@rio.rj.gov.br)

É Analista de Sistemas. Possui 28 anos de experiência como desenvolvedor / gerente de projetos. Responsável técnico pelo projeto Remédio em Casa da SMS-RJ.

Gustavo Boletta (boletta@rio.rj.gov.br)

É Engenheiro de Software. Possui 6 anos de experiência em desenvolvimento de sistemas OO. Trabalha na SMS-RJ há 5 anos como analista responsável pela pesquisa e desenvolvimento utilizando software livre.

Max Moura Wolosker (mwolosker@rio.rj.gov.br)

É Analista de Sistemas. Possui 9 anos de experiência administrando e implantando sistemas baseados em Linux. Trabalha na SMS-RJ há 5 anos como administrador da rede.



Considerações Finais



Os Impactos da Modelagem Conceitual de Dados na qualidade da estrutura de um Banco de Dados

José Alberto F. Rodrigues Filho
beto@beto.pro.br



Piracicaba (SP), Julho de 2005.

Resumo

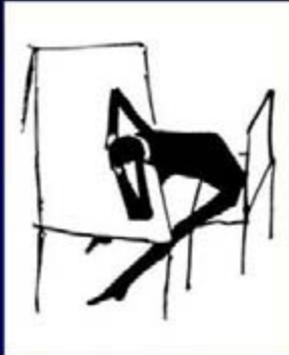
A modelagem de dados é fator de extrema importância para a construção de BDs eficientes e que cumpram o seu papel. A palestra visa discutir a relevância da elaboração de um Modelo Conceitual de Dados como processo importante na construção de um Banco de Dados. Identificar os referenciais apresentados pelo Modelo Conceitual e seus desdobramentos no projeto lógico do Banco de Dados. Avaliar, a partir do Modelo Conceitual, quais as regras de negócio passíveis de efetivação através de restrições (constraints) no Banco de Dados. Discutir o que é “teórico” ou “prático” no contexto de Projeto de Banco de Dados.

Palestrante

José Alberto F. Rodrigues Filho é professor da UNIMEP há 17 anos. Mestre em Engenharia pela POLI/USP, atuando em pesquisas relacionadas à Data Mining e Projeto de Data Warehouse. Adepto do software livre, adota o FireBird como instrumento de aprendizado em suas disciplinas de Banco de Dados.

"All human error is impatience, a premature renunciation of method, a delusive pinning down of a delusion."

Franz Kafka, Letters



Modelagem Conceitual de Dados
Piracicaba (SP), Julho de 2005.



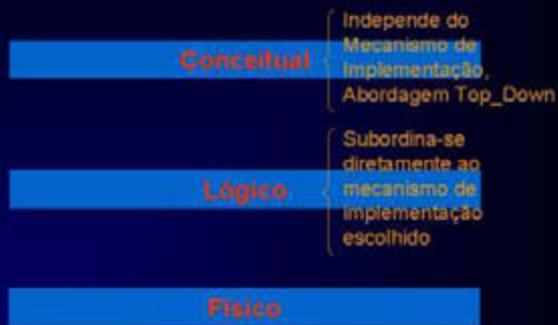
Fluxo do Processamento de Dados



Modelagem Conceitual de Dados
Piracicaba (SP), Julho de 2005.



Níveis de Abstração dos Dados



Modelagem Conceitual de Dados
Piracicaba (SP), Julho de 2005.



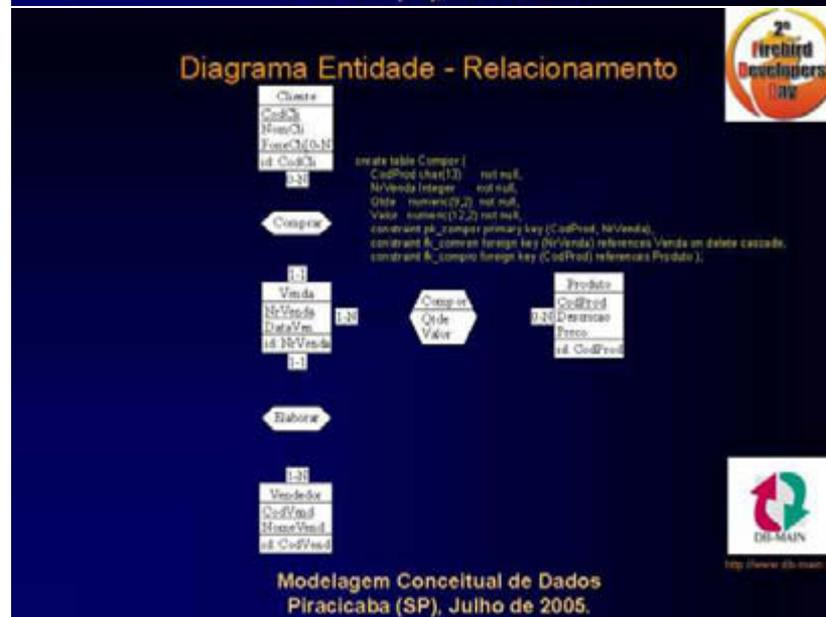
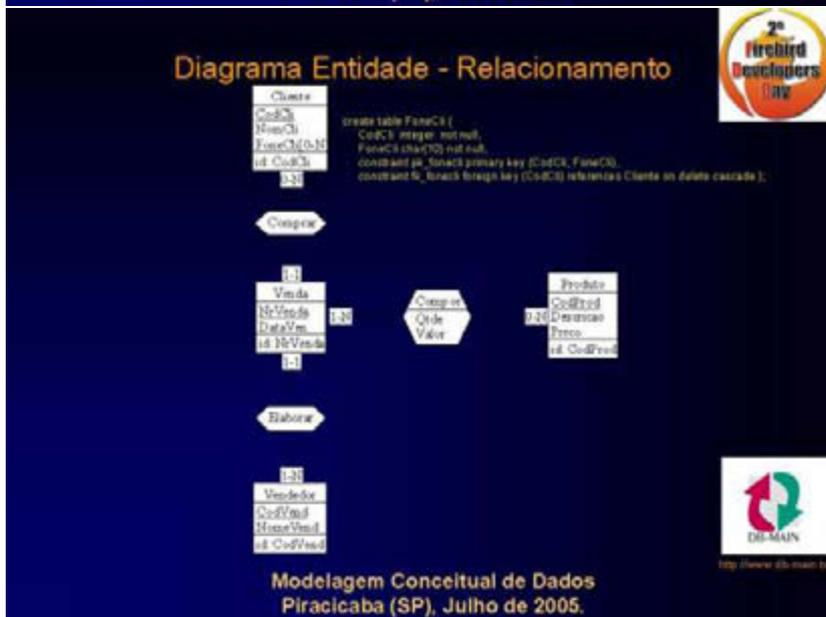
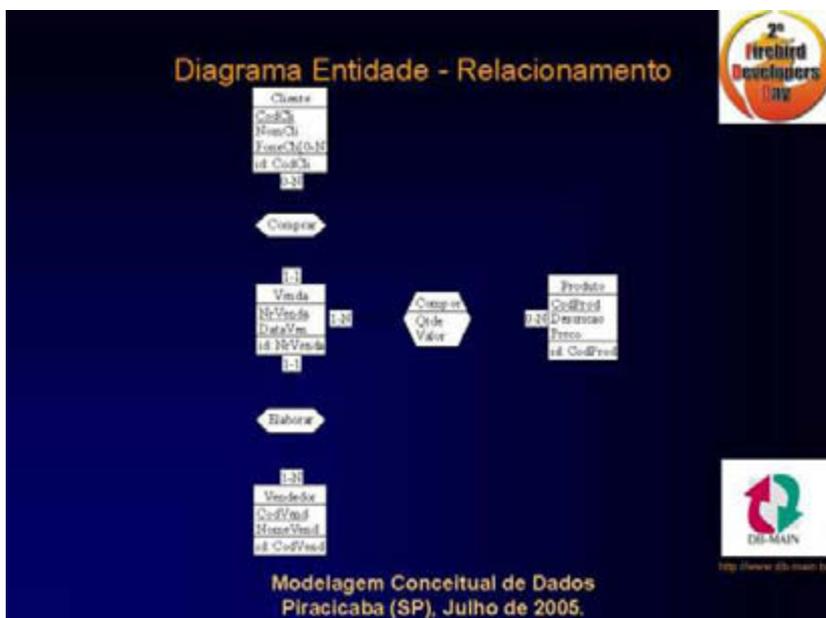
Técnicas de Modelagem Conceitual

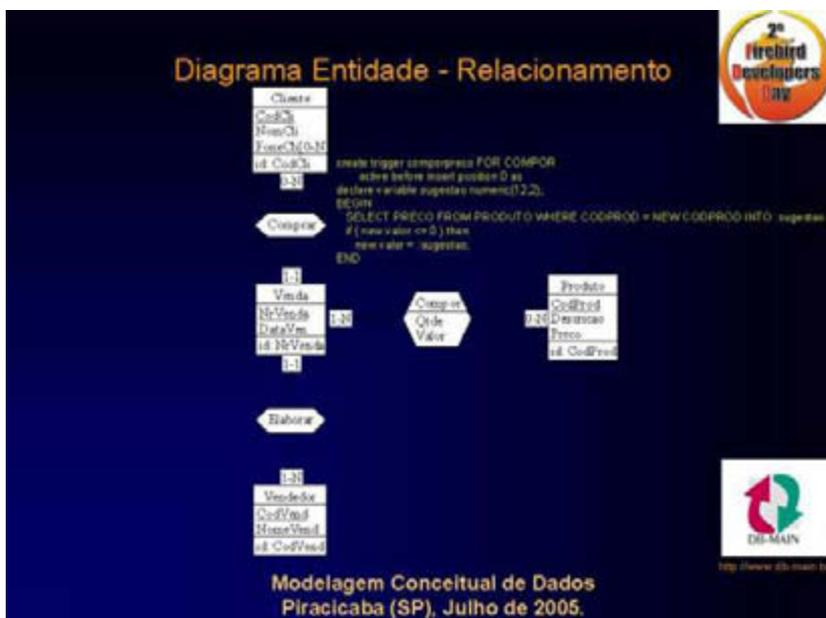
Modelo Entidade Relacionamento

- Desenvolvida por Peter Chen, 1976.
- Técnica de modelagem conceitual mais difundida.
- Representação do modelo através do Diagrama Entidade-Relacionamento.

Modelagem Conceitual de Dados
Piracicaba (SP), Julho de 2005.







Não é proibido intuir apenas não se pode renunciar ao método "desconsiderando-o"

José Alberto F. Rodrigues Filho
 beto@beto.pro.br

Piracicaba (SP), Julho de 2005.

Vox on Demand

Automação de discagem e emissão de recados

URA - Unidade Remota de Atendimento

Gravação de ligações telefônicas

Site: <http://www.jobvox.com.br>
 E-Mail: SAC@JOBVOX.COM.BR

Piracicaba - São Paulo

Precisando de cursos ou treinamentos de Firebird?

A FireBase oferece cursos/treinamentos de Firebird ministrados dentro da sua empresa. Os cursos são ministrados por **Carlos H. Cantu**, um dos maiores evangelistas do Firebird no Brasil, autor do livro Firebird Essencial. Mais informações pelo email cursos@firebase.com.br



Resumo

Todos sabemos que hoje é muito fácil copiar um banco de dados Firebird, levar para outra máquina e acessar as informações que estão nele. Essa palestra mostrará algumas dicas para tornar o servidor Firebird mais seguro, dificultando o trabalho dos intrusos. Serão apresentadas dicas para melhorar a segurança do servidor no Windows 200x/XP e no Linux.

Palestrante

Luiz Paulo de Oliveira Santos, tecnólogo em Processamento de Dados especializado em Análise de Sistemas pela UNIMEP. Atualmente é Analista de Suporte para Internet e Redes Locais da UNIMEP. É proprietário da empresa JobVox e editor da revista DB FreeMagazine

Introdução ao Firebird:

- DBA não requerido
- Alta performance tanto em aplicações críticas como em aplicações menores
- Versioning
- Arquitetura SuperServer
- Instalação em minutos
- Sinalizadores de Eventos
- Funções Definidas pelo Usuário (UDFs)
- BLOBs - Binary Large Objects
- Arrays Multidimensionais
- Banco de dados distribuídos para flexibilidade da aplicação
- Commit em duas fases
- ANSI SQL-92
- Sistema de travamento otimizado
- Usuários de peso
- Flexibilidade de plataformas Windows, Linux, Unix, Solaris, 64, e outros...
- Firebird tem distribuição livre. Não precisa ser comprado



Segurança no hardware:

- Segurança Física para sala de servidores.
- Hardware atualizado e em garantia (on-site se possível).
- Dispositivos de Back Up
- No-Break
- Evitar manter ativo nos servidores dispositivos como Drivers, Cds, DVD, portas USB e outros.
- Manter teclado bloqueado e monitor desligado e console travada com senha



Sistemas Operacionais:

- **Windows**
- **Linux**
- **Outros**



Sistemas de Arquivos:

Windows:

FAT
FAT-32
NTFS

UNIX:

Ext2
Ext3
ReiserFS
NFS

Principais problemas:

- Corrupção de hardware
- Corrupção lógica
- Sub-alocação
- Sistemas de arquivos comprimidos
- Remontagem manual
- Desligamento incorreto
- Buffers grandes

Cuidados:

- Evitar cópias da base
- Evitar manter Back ups antigos



Usuários no Firebird:

Usuários
Finalidades do usuário
Implementação dos usuários
Invasão dos banco através de usuários



ISC4.GDB e SECURITY.FDB:

Conceitos
Concepção
Cuidados
Motivo da externalização dos bancos de segurança
Dicas sobre tamanho da senha e do usuário
Caso LOCKSMITH



Utilitário GSEC:

O que é?
Variáveis ISC_USER e ISC_PASSWORD
Alterar a senha dos usuários
Na versão 2.0 (clausula -server e sem path)



Arquivo Firebird.conf:

Criação
Utilidade para segurança
Arquivo



Arquivo Aliases.conf:



Criação

Utilidade para segurança

Arquivo



Criptografia e Sniffers:



O que é sniffer?

Criptografia para que?

O que é VPN?

Open-VPN

O que é Zebedee?

Criptografia no Zebedee

Compressão no Zebedee

Como funciona?

Compressão de dados

Exemplo



Log de Acesso & Firewall :



Função do Firewall:

Proteger o servidor

Logar os acessos

Barrar intrusos

Softwares:

Linux - IPTables

Windows - ZoneAlarm

Windows - Sygate Personal Firewall



Back up e GBAK :



Para que serve os Back Ups?

Um Back Up pode gerar um banco novo

Confeccionar Back Ups

Ferramenta do Firebird para Back Ups

GBAK - Utilitário de linha de comando



GRANT e REVOKE :

GRANT – Instrução SQL para atribuição de direitos a usuários em um banco de dados SQL.

REVOKE – Instrução que remove direitos atribuídos com GRANT.

Como proteger seu banco de cópia de arquivo usando GRANT e REVOKE.

Dica de Ivan Prenosi:

```
GRANT UPDATE(PASSWD, GROUP_NAME, UID, CHD, FIRST_NAME, MIDDLE_NAME,  
LAST_NAME) ON USERS TO PUBLIC;
```



Noções de Networking para segurança:

A ordem não reflete a real importância...

- 1) Utilizar IPs inválidos (não roteáveis) sempre que possível.
- 2) Redes baseadas em Switches evitam que software sniffer opere.
- 3) Não deixar ativos serviços de controle remoto no servidor.
- 4) Se possível desabilitar ICMP do servidor.
- 5) Não rodar serviço público de arquivos SAMBA ou compartilhamentos nos servidores sempre que possível.
- 6) Evitar rodar serviços como WebServer e FTP no servidor de banco de dados.



Dicas de especialistas:

Evitar o uso de softwares anti-vírus no servidor ou configurá-los corretamente.

Tomar cuidado com o local onde se grava o banco e evitar arquivos externos. Prefira usar aliases.

A melhor política de segurança é aquela que só você conhece!

Cuidado com os usuários e senhas (tamanho de cada um).

Implemente sempre mais que uma rotina de segurança.



Os dez mandamentos da segurança com o Firebird:

1. Não usará jamais a senha **masterkey**, seja para o SYSDBA ou outro usuário.
2. Terá sempre back up atualizado, shadows e base replicada.
3. Usará sempre o Zbedee e um firewall para proteger seu Firebird.
4. Não abusará de U.D.F.s, e, principalmente de UDFs que você não criou!
5. Não utilizará as variáveis ISC_USER e ISC_PASSWORD em vão.
6. Manterá o Hardware e sistema operacional atualizados.
7. Não atribuirá número IP válido ao servidor sem real necessidade.
8. Não deixará faltar espaço no disco rígido do servidor do seu banco de dados.
9. Contribuirá sempre com a Firebird foundation.
10. Lerá a DB Freemagazine mensalmente, o livro do Gantu (o Brasileiro, não o italiano) e o da Helen Bourie também.



Lançamento para Agosto:



Segurança no servidor

- ⇒ Introdução ao Firebird
- ⇒ Segurança no hardware
- ⇒ Sistemas Operacionais
- ⇒ Sistema de Arquivo
- ⇒ Usuários no Firebird
- ⇒ isc4.gdb e security.fdb
- ⇒ Utilitário GSEC
- ⇒ Arquivo: firebird.conf
- ⇒ Arquivo: aliases.conf
- ⇒ Criptografia e *sniffers*
- ⇒ Log de Acesso e Firewall
- ⇒ BackUp e GBAK
- ⇒ Utilitário GFIX
- ⇒ GRANT e REVOKE
- ⇒ Noções de Networking
- ⇒ Dicas de especialistas
- ⇒ Os 10 mandamentos
- ⇒ Firebird – Dicas de Seg.



Segurança no servidor
Luiz Paulo de Oliveira Santos



Próxima palestra às 13h10
Novidades do Firebird 2.0
Carlos Henrique Cantu

Sobre minha palestra...

Um dos tópicos que mais chamou a atenção do público em minha palestra foi a utilização do Zebedee em conjunto com Firebird.

Logo, gostaria de deixar essa dica: O uso do Zebedee (tunel TCP criptografado) é um meio seguro, de fácil configuração, e com significativo ganho de banda passante, pois compacta os dados antes de enviá-los. Logo, links que podiam estar saturados passarão a responder melhor com essa solução gratuita, que funciona em diversos Sistemas Operacionais, entre eles o Microsoft Windows e o Linux.

Outro fator importante é manter uma réplica ativa e atualizada de seus bancos de dados fora de suas instalações, pois em caso de tragédias (roubo, incêndios ou outros) poderá rapidamente reaver suas informações.

Como não há muitos provedores de hospedagem que disponibilizam o Firebird, indico um parceiro que pode auxiliá-los nessa tarefa: falo do provedor BAVS, que pode disponibilizar Firebird à preços realmente baixos, e Zebedee para Firebird gratuitamente para seus assinantes. O BAVS possui um serviço profissional e sério, com ótimo suporte. Entre em contato pelo link: www.bavs.com.br. E lembre-se utilize sempre o Firebird com Zebedee.

Até a próxima edição.
Luiz Paulo



Resumo

Muitos dos usuários do Firebird desconhecem ou não entendem o conceito e o funcionamento da arquitetura de Versioning do banco, responsável pelo controle de concorrência. A correta utilização do controle transacional é fator chave para a performance de diversas operações, e consistência das informações no BD. Essa palestra mostrará na prática o que são e como funcionam as transações no Firebird, seus modos de isolamento e suas aplicações.

Palestrante

Eduardo Mercado trabalha com desenvolvimento de softwares desde 1990, é gerente de projetos da Young Brasil Consultoria, instrutor Delphi, Kylix e Firebird. Trabalha com Delphi desde a versão 2, e com o Kylix desde sua primeira versão. É autor do livro “Kylix - Delphi para Linux com Interbase e Firebird”, colunista da revista ClubeDelphi e ministrou várias palestras sobre Delphi, Kylix, Firebird e .NET

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Transações

EMISSÃO NOTA FISCAL

Transação

Gravação dos itens da nf
Baixa de estoque
Geração de comissões
Geração de duplicatas

Gravação efetiva do pacote
COMMIT

Cancelamento da operação
ROLLBACK

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Transações

- O Firebird permite várias transações ativas ao mesmo tempo.
- Permite também criar uma única transação relacionada a bancos diferentes (transações distribuídas).
- Erro DeadLock.

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Versioning

- Travamento otimista
- Backup
- Log de transações

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Comandos de transação

- StartTransaction – inicia uma nova transação.
- Commit – efetiva a gravação do pacote gerado na transação.
- CommitRetaining – efetiva a gravação do pacote gerado pela transação, e mantém a transação ativa.
- Rollback – desfaz o pacote gerado pela transação.
- RollbackRetaining – desfaz o pacote gerado pela transação, e mantém a transação ativa.

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Isolamento Transacional

- Read Committed
 - Visualizam os dados efetivamente gravados (commit) por outras transações.
- SnapShot
 - Visualizam os dados espelhados no momento em que a transação foi iniciada.
- Snapshot table stability
 - Possui as mesmas características do isolamento Snapshot, mas depois que um tabela é lida não pode ser alterada por nenhuma outra transação.

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Travamento Pessimista

- Cláusula With Lock
- Cláusula For Update

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Modos de bloqueio

- Wait – a segunda transação gerada aguarda a finalização da primeira quando ocorre um erro de deadlock.
- NoWait – erros de deadlock são informados no instante em que o problema ocorre.

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Componentes responsáveis por transações no Delphi

- DbExpress – SqlConnection ...
- IBX – IbTransaction ...
- IBO – Ib_Transaction
- BDP - BDPCConnection

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Cuidados com tratamento de cache

- Vale salientar que os dados só estão contidos na transação no momento que são enviados ao banco de dados, ou seja, quando se utiliza recursos de cache com um clientdataset por exemplo, os dados só estão na transação depois do applyupdates no dataset relacionado.

Eduardo Mercado – Young Brazil Consultoria e Treinamento

Tratamento de transações Firebird

Eduardo Mercado
 Young Brazil Consultoria e Treinamento



SQL> SELECT * FROM HOSPEDAGEM WHERE QUALIDADE="INSUPERAVEL";

+-----+
 | WWW.BAVS.COM.BR |
 +-----+

MySQL Firebird

+ PLANOS

acesse:
WWW.BAVS.COM.BR

+ INFORMAÇÕES

clientes.com.satisfação
 Email: info@bavs.com.br
 Atendimento Eletrônico 24hrs:
 (19) 3421-0251
 Vendas On-Line (mediante segurança):
 http://www.bavs.com.br

PLANOS DE HOSPEDAGEM		
PRO I	PRO III	SEMI D. II
100 Mb espaço em disco Firebird 1.0 MySQL 3.23 PHP 4 Perl 5 CGI Diretório SSL Gratuito Configuração Gratuita Mensalidade: R\$ 29,90	300 Mb espaço em disco Firebird 1.5 MySQL 4 PHP 4 / Perl 5 / CGI JSP (Tomcat) Servlet ASP .NET (Mono C#) Configuração Gratuita Mensalidade: R\$ 69,90	1 Gb espaço em disco Firebird 1.5 MySQL 4 PHP 4 / Perl 5 / CGI JSP (Tomcat) Servlet ASP .NET (Mono C#) Configuração Gratuita Mensalidade: R\$ 145,90

The slide features a large title 'WebServices' at the top. Below it, a green horizontal bar contains the name 'emerson FACUNTE' in yellow and white, with 'co-Owner Framework .Net Microsoft Latam' underneath. To the right is a circular logo for '2º Firebird Developers Day'. At the bottom left is the 'WebServices' logo, and at the bottom right are the 'Facunte .net' and 'DevMedia' logos.

Resumo

O mercado está se rendendo ao mundo dos WebServices. Os sistemas operacionais estão mais preparados para a tecnologia. Os profissionais estão cada vez mais qualificados. Mas e os bancos de dados, como estão? O Firebird vai muito bem neste segmento, e com o seu forte suporte a múltiplas conexões simultâneas, tem conquistado um grande número de adeptos. Nesta sessão, ilustrarei a parte de negócios dos WebServices, bem como o desenvolvimento passo-a-passo utilizando ferramentas .Net (Delphi 2005, C# Express, Visual Studio 2005), entre outras.

Palestrante

Emerson Facunte é co-Owner Framework.Net Microsoft Latam, evangelista de aplicações e-business, publicou 6 livros e mais de 80 artigos, ministrou palestra para cerca de 10.000 pessoas em todo o país (Borcon, TechWeek-SP-POA-RJ, .Net 2005 RoadShow), membro-fundador do DUG-BR, líder do grupo Go.Net, consultor sênior do grupo DevMedia e Arquiteto de Software do Grupo Saraiva. Nos tempos livres curte programação MSX, Cinema, Basquete e uma boa pista de dança!

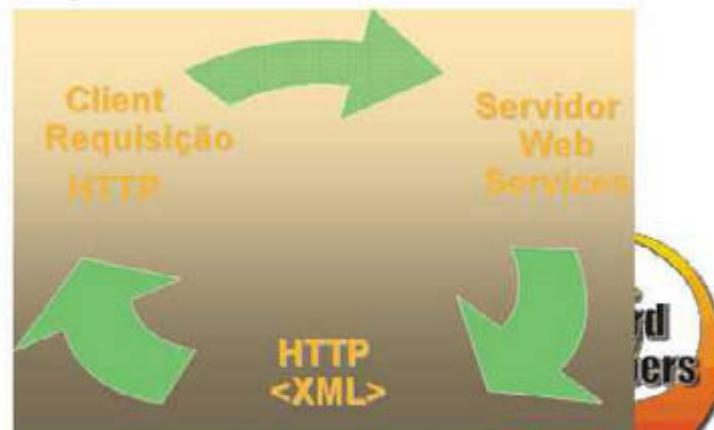
Overview

- ◆ Arquitetura
- ◆ Aplicabilidades
- ◆ Segurança
- ◆ Quem já implementou
- ◆ Demo
- ◆ Questions!

WebServices



Arquitetura



WebServices



Aplicabilidades

- ◆ Troca de informações entre parceiros
- ◆ Integração de sistemas
- ◆ Consulta documentos
- ◆ Vendas
- ◆ Mobilidade

WebServices



Segurança

- ◆ HTTPS (SSL)
- ◆ wsSecurity (nivel 2)
- ◆ Sessões

WebServices



Quem já implementou

- Saraiva
- Fedex
- Buscape, Jacotei, Farejador
- Itau
- Bradesco
- Caixa Economica
- Banco do Brasil
- Economatca
- Ernst & Young
- Google
- DHL



WebServices

Facunte DevVulcan

Demonstração

Let's Go Delphi Rulez!!!



WebServices

Facunte DevVulcan

Sala de Interrogatório



WebServices

Facunte DevVulcan

Novidades do Firebird 2.0



www.FireBase.com.br

Carlos H. Cantu (2005)

www.FireBase.com.br

Resumo

Conheça os principais novos recursos que o Firebird 2.0 traz para você: Execute Block, novos collates, cursores em PSQL, tabelas derivadas, protocolo XNet, Sequences, Rows, Cross Joins, variáveis contextuais, IIF, etc.

Palestrante

Carlos H. Cantu - WarmBoot Informática/FireBase - atua no mercado de desenvolvimento de software há 15 anos, é mantenedor do site www.firebase.com.br (referência nacional do IB/FB), consultor de banco de dados Firebird e InterBase 6.0, autor do livro Firebird Essencial, editor da revista DBFreeMagazine, e desenvolvedor Delphi desde a versão 1.0.

Tabelas derivadas (aka. Select from select)

- Permite que um select seja a fonte de dados para um outro select
- As colunas recuperadas pela query derivada devem possuir nomes ou alias definidos, caso contrário, deve-se nomear as colunas da tabela derivada
- Exemplo:
Select avg(salario) as media_salario
from (select first 10 f.salario
from funcionarios f
order by f.salario desc)

www.FireBase.com.br

Conexões locais com o servidor Windows

- Novo protocolo **XNET**
- Pode ser usado no Firebird Classic
- Acaba com o problema da serialização da execução de instruções encontrado no protocolo antigo

www.FireBase.com.br

LIKE/CONTAINING/STARTING WITH

- Funções re-implementadas
- Funcionam corretamente com blobs do tipo texto
- Utilizam agora o algoritmo Knuth-Morris-Pratt (mais eficiente)
- Não “quebram” mais o servidor quando NULL é usado com LIKE

www.FireBase.com.br

Novos modos de ShutDown

- **Single:** Apenas uma conexão SYSDBA pode existir – útil para rodar scripts de atualização de metadata, etc.
- **FULL:** Nenhuma conexão é aceita – útil para realizar cópias físicas dos bancos, etc.
- **Multi:** modo presente nas versões < FB 2.0 onde várias conexões SYSDBA são aceitas.

www.FireBase.com.br

NULLs em UDFs

- Agora é possível passar e identificar NULLs em UDFs sem a necessidade de usar descritores
- Deve-se incluir a palavra NULL na declaração dos parâmetros que poderão ser nulos, ex:

```
declare external function f_left  
cstring(255) NULL  
returns cstring(255) free_it
```

- A UDF deve ser escrita de forma a saber interpretar essa nova possibilidade

www.FireBase.com.br

Aliases com GROUP BY e ORDER BY

- Os comandos **Group By** e **Order By** agora podem especificar as colunas a serem processadas através do seu *alias* previamente definido, ex:

```
Select cidade, count(*) as total  
from clientes  
group by cidade  
order by total desc
```

www.FireBase.com.br

EXECUTE BLOCK

- **Sintaxe:**
EXECUTE BLOCK [(param datatype = ?, param datatype = ?, ...)]
[RETURNS (param datatype, param datatype, ...)]
AS
[DECLARE VARIABLE var datatype; ...]
BEGIN
...
END
- Permite executar blocos de comandos como se fossem procedures, sem no entanto ter que criá-los previamente no banco
- Útil para rotinas que serão executadas uma única vez, como por exemplo atualizações de dados, etc.

www.FireBase.com.br

ROWS

- Semelhante ao FIRST/SKIP
- Compatível com o SQL standard
- Não pode reproduzir o uso do SKIP sozinho
- **Exemplo:**
**Select * from clientes
order by nome
rows 1 to 10
-- recupera os 10 primeiros registros**

www.FireBase.com.br

CREATE SEQUENCE

- Sinônimo para GENERATOR
- Compatível com SQL-99
- Sintaxe:

```
CREATE { SEQUENCE | GENERATOR } <name>  
DROP { SEQUENCE | GENERATOR } <name>
```

```
SET GENERATOR <name> TO <start_value>  
ALTER SEQUENCE RESTART WITH <start_value>
```

```
GEN_ID (<name>, <increment_value>)  
NEXT VALUE FOR <name> (*)
```

Next Value FOR incrementa o generator em uma posição.

www.FireBase.com.br

CROSS JOIN

- Permite cruzamento carteziano de duas tabelas.

- Exemplos:

```
Select * from A CROSS JOIN B
```

equivale a ...

```
Select * from A INNER JOIN B ON 1 = 1
```

equivale a ...

```
Select * from A, B
```

www.FireBase.com.br

IIF

- Sintaxe:
IIF (expressao, valor1, valor2)

Se o resultado de "expressao" for verdadeiro, então valor1 é retornado, caso contrário valor2 é retornado

- É um atalho para
CASE
WHEN <expressao> THEN <valor1>
ELSE <valor2>
END

www.FireBase.com.br

Ordem dos NULLs

- O resultado segue o padrão do standard SQL
- Se a ordenação for ascendente, os nulos virão por último
- Se a ordenação for descendente, os nulos aparecerão primeiro

** Aplicável somente em BD's com ODS >= 11*

www.FireBase.com.br

Vários

- RECREATE EXCEPTION e CREATE OR ALTER EXCEPTION
- DISTINCT (trata NULL = NULL como TRUE)
- FIRST/SKIP, ROWS, PLAN e ORDER BY podem ser definidos em VIEWS
- UNION em *subqueries* e *insert into select*
- UPDATE e DELETE com PLAN e ORDER BY
- ROW_COUNT retorna número de registros retornados por um SELECT
- CURRENT_TIMESTAMP e NOW retornam milisegundos

www.FireBase.com.br

Vários (2)

- Possibilidade de armazenar bancos de dados em dispositivos raw (sem sistema de arquivos) no unix e derivados
- Suporte total a API de serviços no servidor Classic, tanto no Windows como no Linux
- Timeouts em transações do tipo WAIT
- Nova ODS (11)
- Exceções podem ter mensagens até 1021 bytes
- Possibilidade de incluir descrições em generators e roles
- Suporte a plataformas de 64bits
- REVOKE ADMIN OPTION FROM
- Melhorias no otimizador de queries

www.FireBase.com.br

Vários (3)

- **Garbage collection** cooperativo + background
- Inteligência na determinação dos tipos dos campos entre os selects de uma UNION
- COMMENT

www.FireBase.com.br

Table Aliases

- A definição de aliases em selects é restritiva. Caso um alias seja definido para uma tabela, não podemos mais utilizar o nome da tabela na indicação do campo.
- `select clientes.nome
from clientes c -- inválido!`

- `select c.nome from clientes c
select nome from clientes c`

www.FireBase.com.br

Cursores explícitos em PSQL

- Definição e controle total de **cursores** em PSQL.
- Permite **updates e deletes** com a cláusula **WHERE CURRENT OF**
- **DECLARE [VARIABLE] <cursor_name> CURSOR FOR (<select_statement>);**
OPEN <cursor_name>;
FETCH <cursor_name> INTO [, ...];
CLOSE <cursor_name>;

www.FireBase.com.br

Cursores explícitos em PSQL (Exemplo)

```
DECLARE CLINOME VARCHAR(50);
DECLARE C CURSOR FOR (
    SELECT NOME FROM CLIENTES);
BEGIN
    OPEN C;
    WHILE (1 = 1) DO
        BEGIN
            FETCH C INTO :CLINOME; -- recupera registro
            IF (ROW_COUNT = 0) THEN
                LEAVE; -- acabou o select... Sai fora.
            SUSPEND; -- retorna informação
        END
    END
    CLOSE C;
END
```

www.FireBase.com.br

LEAVE

- Permite a utilização de "labels" para determinar para onde o código deve seguir
- Semelhante ao "velho" GOTO do BASIC
- Quando usado sem um label definido, determina o encerramento do loop atualmente executado

www.FireBase.com.br

Índices

- Limite de **252 bytes** por chave removido
- Limite atual é aprox. $\frac{1}{4}$ do tamanho da página
- Índices com expressões, ex:
CREATE INDEX IDX_NOME_COMPLETO
ON TAB_PESSOAS
COMPUTED BY (NOME || SOBRENOME);
- Perda de performance em índices com alta taxa de valores repetidos não existe mais!

www.FireBase.com.br

Otimizador

- Muitas melhorias no otimizador de queries
- Planos mais inteligentes
- Performance superior com IN e OR

www.FireBase.com.br

NBACKUP

- Possibilidade de criar backups auto-incrementais
- Opções:
 - L: Trava o banco de dados para copia do arquivo via sistema operacional
 - U : Destrava um BD
 - F: Destrava a cópia do BD
 - B <level> <database> [<filename>]: Cria um backup incremental
 - R <database> [<file0> [<file1>...]]: Restaura um backup incremental
- Backups de BDs "multi-arquivos" não são suportados

www.FireBase.com.br

Segurança

- Algoritmo de hash para armazenamento de senhas melhorado
- Usuários podem modificar suas próprias senhas
- Acesso ao banco de dados de segurança (security.fdb) somente através da API de serviços.
- Proteção contra ataques de "força bruta"
- Script *security_database.sql* para atualizar o security.fdb antigo
- Parâmetro **LegacyHash** no *firebird.conf* liga e desliga o novo sistema de segurança

www.FireBase.com.br

Variáveis contextuais

- Permite a leitura e gravação de valores em variáveis associadas à namespaces através das funções internas:
`RDB$SET_CONTEXT(<namespace>, <variable>, <value>)`
`RDB$GET_CONTEXT(<namespace>, <variable>)`
- Namespaces definidos automaticamente:
USER_SESSION
USER_TRANSACTION
SYSTEM
 - NETWORK_PROTOCOL – TCPv4, WNET, XNET e NULL
 - CLIENT_ADDRESS – endereço IP da conexão do cliente, o ID de uma conexão local (XNET) e NULL caso seja outro protocolo.
 - DB_NAME – Path do banco de dados, ou alias.
 - ISOLATION_LEVEL - READCOMMITTED, CONSISTENCY, SNAPSHOT
 - TRANSACTION_ID – similar a CURRENT_TRANSACTION.
 - SESSION_ID – similar a CURRENT_CONNECTION.
 - CURRENT_USER – similar a CURRENT_USER ou USER.
 - CURRENT_ROLE – similar a CURRENT_ROLE.

www.FireBase.com.br

Novo módulo INTL (1)

- **Upper** com caracteres acentuados
- Checagem do tamanho **lógico** novos **charsets multibyte**
- **Collates** podem ser utilizados na recuperação de **blobs**
- **UTF8** no lugar de **UNICODE_FSS**
- Collates **case/accent insensitive**:
 - PT_BR / WIN_PTBR (WIN1252)
 - PT_BR (ISO8859_1)

www.FireBase.com.br

Novo módulo INTL (2)

- **Novas funções para strings**:
 - TRIM
 - LOWER
 - BIT_LENGTH
 - CHAR_LENGTH/CHARACTER_LENGTH
 - OCTET_LENGTH

www.FireBase.com.br

FIM

FIM

www.FireBase.com.br

Patrocinadores do 2º FDD:



Database Corruption – Identify it, Correct it, Prevent it

Ann Harrison
IBPhoenix
aharrison@ibphoenix.com



Resumo

Quando problemas acontecem, os bancos de dados podem se corromper fisicamente ou logicamente. Essa palestra descreve as diferenças entre os dois tipos de corrupção, como cada um deles pode acontecer, como evitar que isso ocorra, diagnosticar, e corrigir os problemas quando for possível.

Palestrante

Ann Harrison trabalha com Bancos de Dados há mais tempo do que gostaria de admitir, iniciando com a DEC. Desenhou também um dicionário de dados baseado no modelo de dados semântico, criando um protótipo usando Rdb/ELN. Ann trabalhou para a InterBase (ISC) e desde então vêm acompanhando o desenvolvimento do InterBase e do Firebird, ao lado de Jim Starkey. Atualmente é responsável pela IBPhoenix USA.

Overview

Types of corruption

Detection

Causes

Recovery

Prevention



Types of corruption

Logical

Nulls in not-null field

Duplicates in unique field

Referential integrity

Physical

Orphan pages

Wrong page type



Logical corruption - detection

Restoring a backup fails



Logical corruption - causes

Firebird does not validate constraints on definition

Except – Unique & Referential

Example – null in not null field

– invalid constraint

Hard crash with forced writes off



Logical corruption - recovery

Restore backups periodically using `-v` switch
Identify and correct problems in the database
IBBackupSurgeon



Logical corruption – prevention

Use forced writes or UPS
Validate data before adding constraints



Physical corruption - detection

Errors in the Firebird log
 Invalid page type expected (x) encountered (y)
 Orphaned back version
 Index errors
Program errors
 Same text
Backup – not restore – fails



Physical corruption - causes

Hard crash with forced writes off

Bugs in Firebird
 Report errors to the support list first
 If we don't know about it, it won't be fixed



Physical corruption - recovery

Depends on the error...

- Wrong page type (expected 7 found 5)
- Orphan pages
- Orphan back versions

Metadata backup & reload database



Physical corruption – prevention

Forced writes or UPS

Firebird 1.5 and Firebird 2.0 are better on Windows



Internal gds error (can't continue after bugcheck)

Always follows another error

Indicates possible data structure corruptions –
or lazy programmers

Prevents further damage

Does not help diagnose error



Summary

Use forced writes or UPS

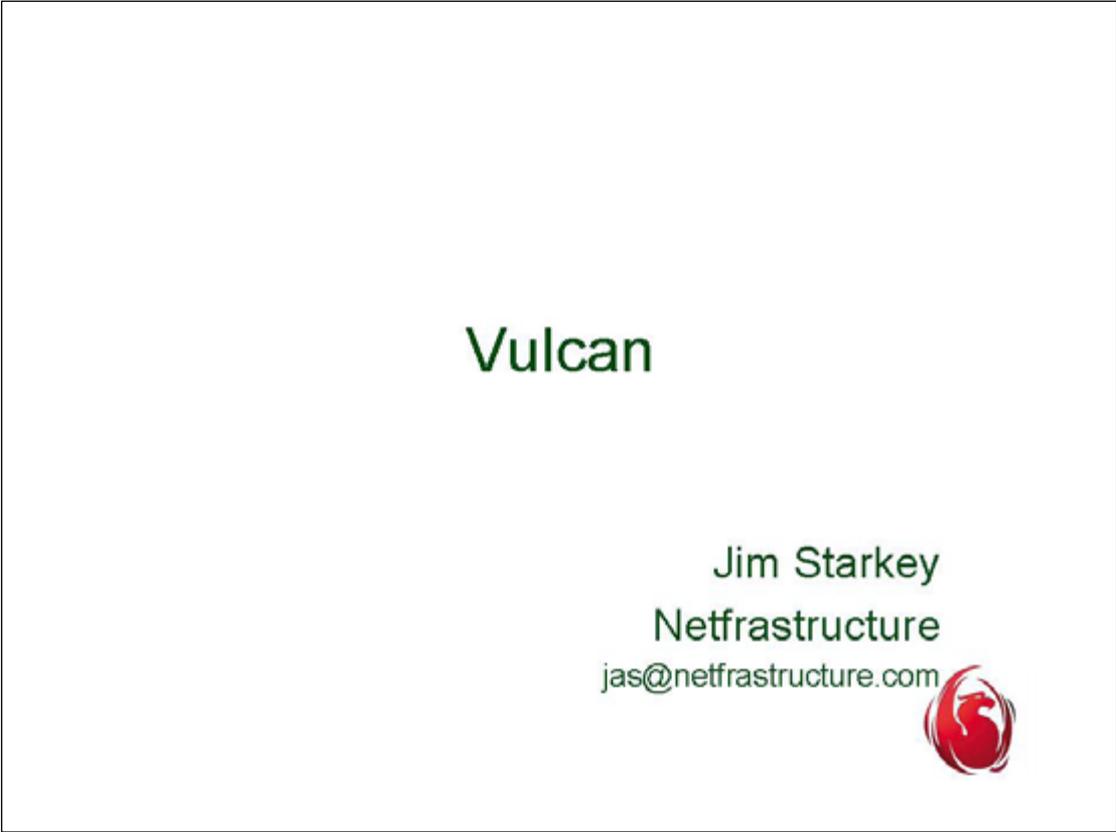
Validate data before adding constraints

Use forced writes or UPS

Upgrade to FB1.5

Use forced writes or UPS





Resumo

Com a aproximação do lançamento do Firebird 2.0, o foco dos desenvolvedores começa a se voltar para o Vulcan. Originalmente desenvolvido por Jim Starkey, para atender a necessidade de um servidor compatível com SMP, o Vulcan agora representa o futuro do Firebird. Além de ser um servidor com ajuste fino de granularidade em multitarefa, o Vulcan separa as funções do Firebird em Servidor, Interface remota, Engine e extensões futuras, como sistema de gateways externos. O refactoring do código propiciou a remoção de muito código condicional, e permitiu que as aplicações determinassem em tempo de execução o tipo de arquitetura a ser utilizada: Classic, Shared Server, Shared Cachê ou Embedded.

Palestrante

Jim Starkey é o inventor do InterBase (1984), e idealizador da arquitetura de versioning. Antes disso, Jim trabalhou para a DEC (Digital Equipment Corporation), onde escreveu uma linguagem para queries chamada Datatrieve, designou a arquitetura corporativa do Banco de Dados, e liderou o projeto que criou o Rdb/ELN, na sua primeira implementação. Após a venda do InterBase (1991), Jim criou uma empresa que inventou uma linguagem para queries gráfica e independente de sintaxe. Atualmente está desenvolvendo uma aplicação de última geração para desenvolvimento com banco de dados/web chamada Netfrastructure, e é o principal desenvolvedor do Firebird Vulcan, que se tornará o Firebird 3.0

Vulcan – Sponsor’s Requirements

64 Bit (Solaris UltraSparc)

SMP Friendly

At least 3X performance on 4 processor system

Single thread performance with 20% of
Firebird 1.5



Vulcan – My Requirements

Strict compatibility with Firebird

32 and 64 Bit (AMD64 Linux)

Unified Engine (combine classic, Superserver, and
embedded)

Establish Provider Architecture

Move SQL into the engine



My Requirements

Fine granularity threading

Per-database configuration system

Start transition to portable object oriented technology

Transparent Gateway to legacy Firebird, Interbase



Vulcan – Results

Single thread performance exceeds Firebird 1.5 and 2.0
pre-alpha

Single process, multi-thread performance > 3.5X on 4
processor system

Code base compiles on:

Visual Studio Version 7 for Win32

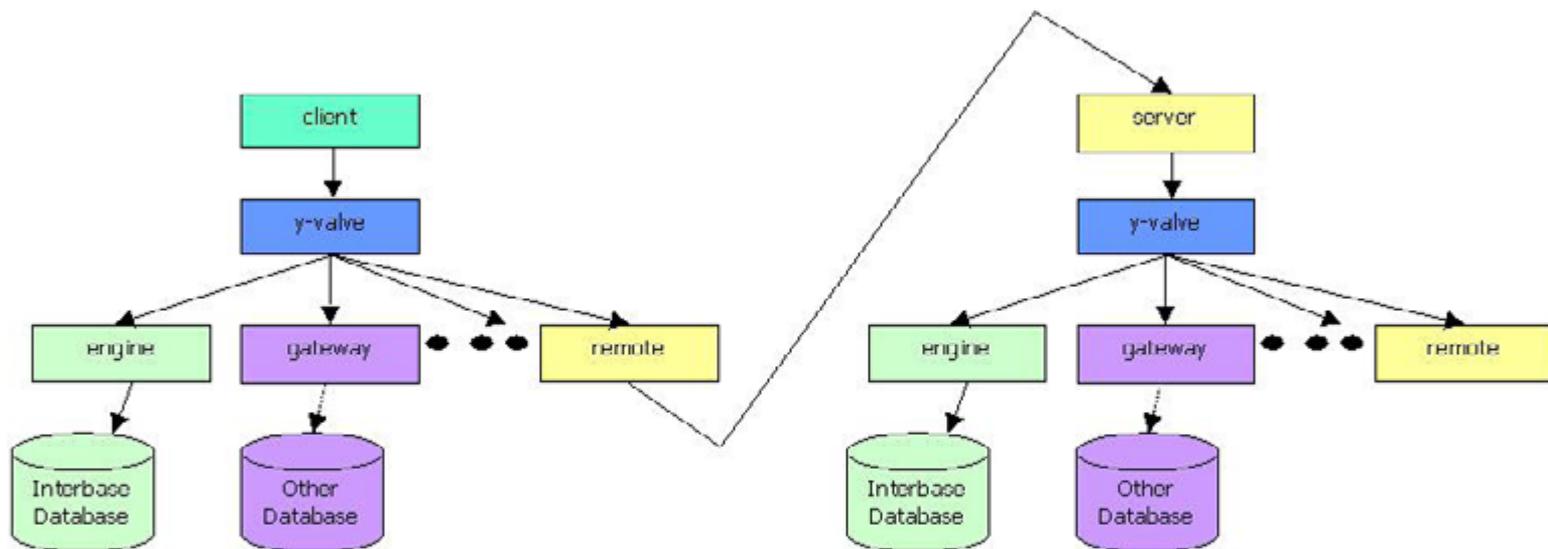
32 bit IA Linux from Red Hat 7.2 to SuSE 9.2

64 bit AMD64 Linux (SuSE 9.2)

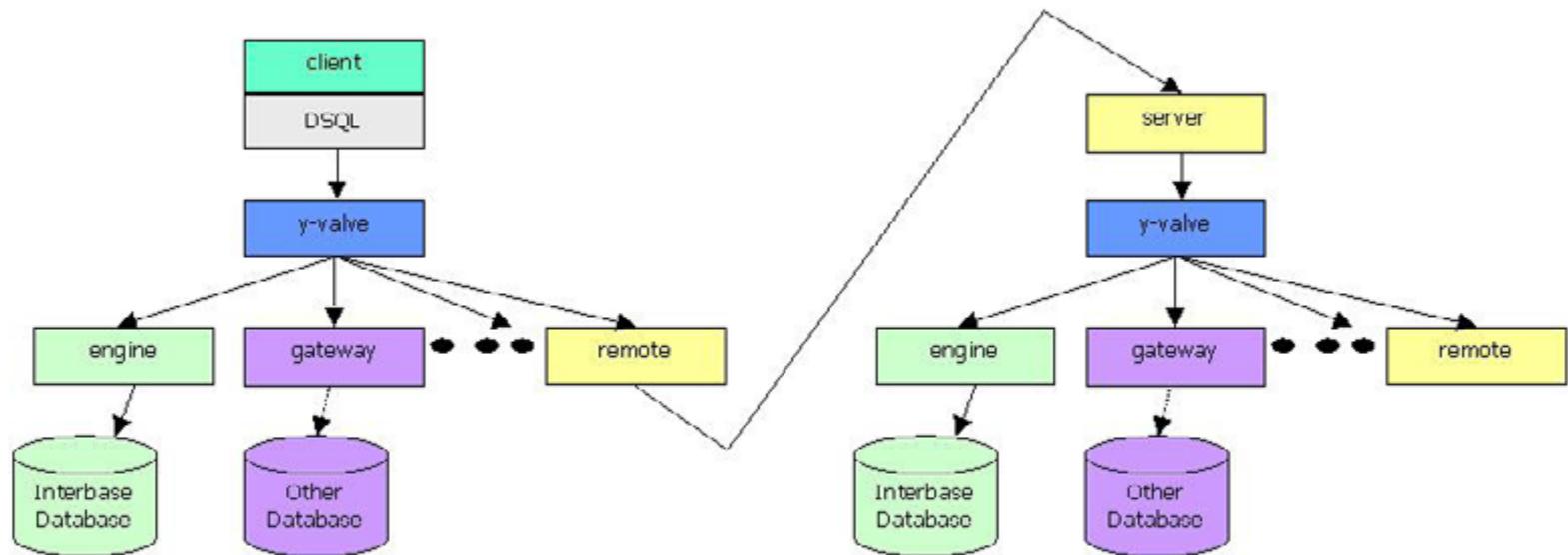
64 bit Solaris with Forte Compiler



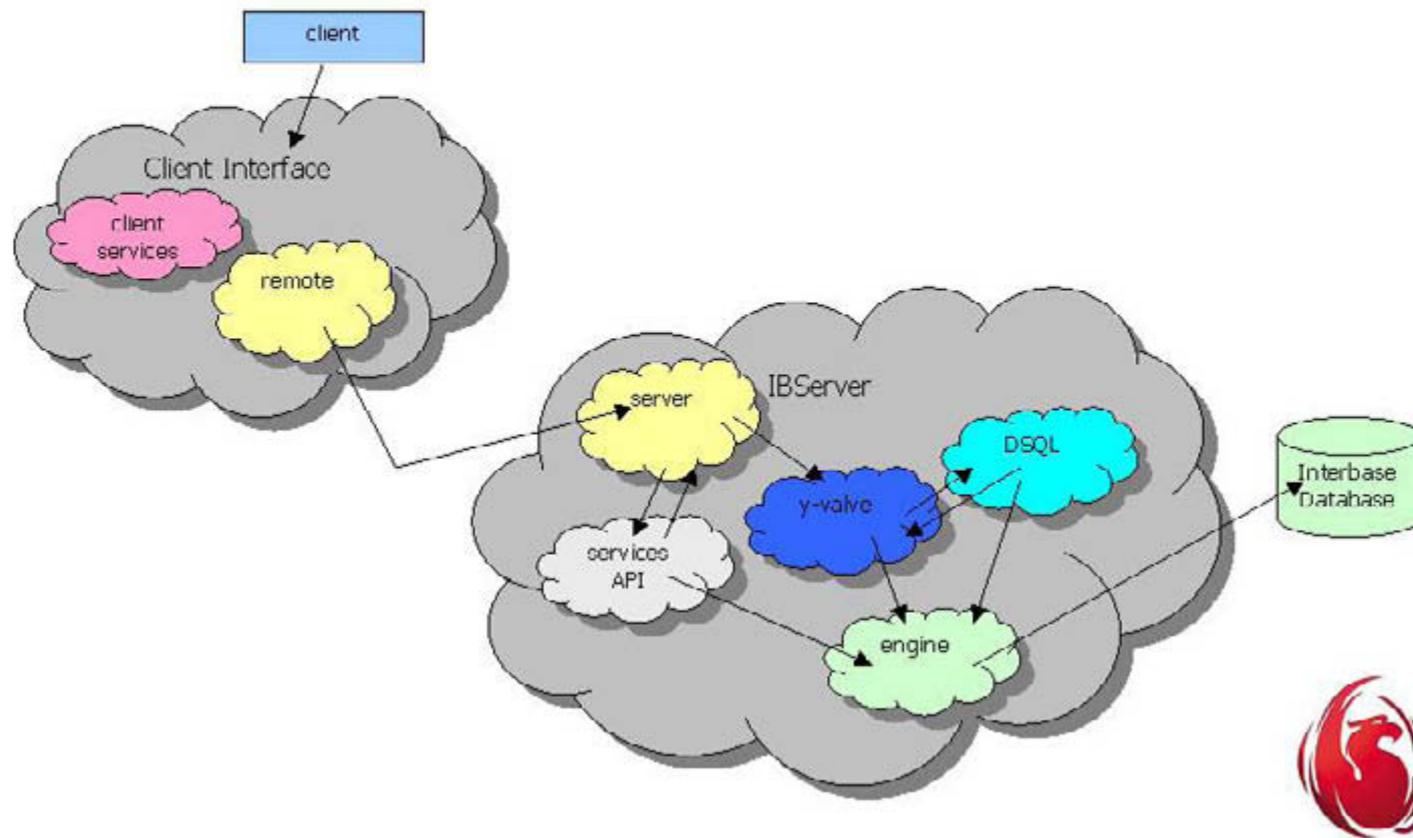
Product Architecture – Original



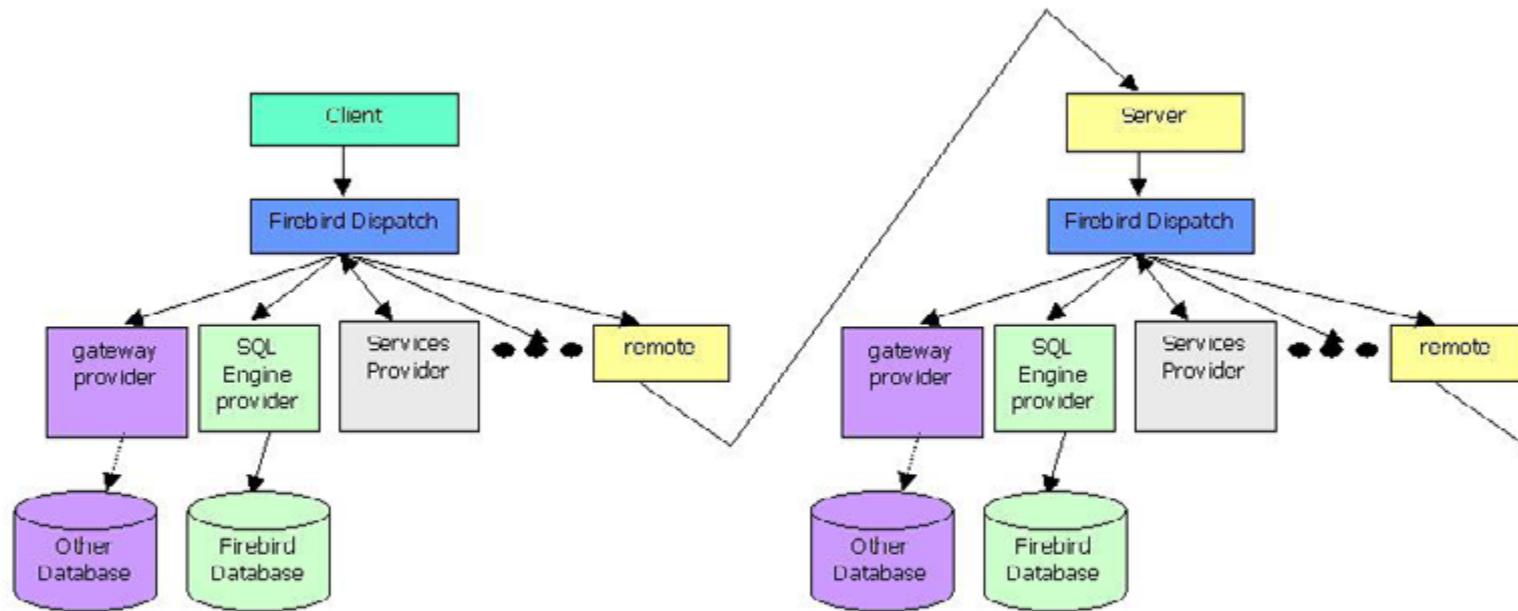
Product Architecture – SQL



Product Architecture – Post Borland



Product Architecture – Vulcan



Vulcan Providers – Abstract Data Managers

A provider a loadable library that implements the provider API

Examples of provider:

- The current engine
- The engine from the prior release
- The remote interface for server access
- The gateway to pre-Vulcan Firebird
- The gateway to current and legacy Interbase

Provider selection is automatic under configuration control



Firebird Engine Builds

Super server – single process, multi-threaded, single page cache

Classic – multiple processes, each with separate page cache

Embedded – single process, no sharing



Vulcan Engine

Per-database configuration determines:

- Local engine, shared server, or gateway to legacy system
- Location of server
- Whether local engine shares database file (embedded or classic mode)
- Whether server shares database file (classic or Superserver)

Single client library for all options

Providers (engines, gateways, remote interfaces) are loaded on demand



Vulcan Security

Each database can

- Use named security database
- Function as own security database
- Have security disabled

New DDL statements to create, alter, and drop users

New security API to propagate user data

Security plugin architecture for extensible security control



Major Internal Changes

Re-established separation of Y-valve and engine
SQL moved from Y-valve to engine
Architectural support for compiled statement cache
Transformation internal record streams to type hierarchy is underway



Major Internal Changes

Object encapsulation of internal mechanisms include:

- Database page cache managers
- Lock Manager
- Database Managers
- SQL Parser
- Compiled requests
- Security Manager



Vulcan – What's Missing

- The services provider
- Miscellaneous loose ends (events, execute statement for example)
- An installation procedure
- Documentation



Vulcan Roadmap

A full functionality beta, Firebird 2.0 compatible
An initial release, Firebird 2.0 compatible, with little new functionality
A follow on release to exploit new internal architecture



Likely Vulcan Futures

- New record encoding
- Internal switch to UTF-8 from multiple character sets
- Elimination of physical field lengths and, maybe, logical field lengths
- New low level API and wire protocol
- Additional high level, object oriented API



Summary

Vulcan has

- SMP & 64 bit support
- Internal SQL
- Solid architecture

Vulcan will have

- All V2.0 Features
- Much more...



Performance hints for Firebird

Ann Harrison

IBPhoenix

aharrison@ibphoenix.com



Overview

Normalization
Indexes
Optimizing inserts
Transactions



Normalization

Create simplest tables that correctly represent the data
More small, simple tables
Reduce data replication
Simplify updates
Can be overdone
"A Simple Guide to Five Normal Forms in Relational Database Theory"
William Kent, 1982 <http://www.bkent.net/Doc/simple5.htm>



Indexes

All fields used in ON and WHERE clauses
Except:
Fields with bad selectivity



Compound indexes

What are they
Index on fieldA, fieldB, fieldC
And fieldB, fieldA, fieldC
And fieldC, fieldA, fieldB and...
Avoid them
Firebird uses multiple indexes
Except
When needed for selectivity



Directional indexes

Indexes locate records

Except

Select max (x)

Select first (n) <fields> from <selection>
order by descending <fields>



Using transactions intelligently

Commit & Commit retaining

Write all data

Write all indexes

Write all internal bookkeeping pages

Write transaction inventory page

Write header page

Don't commit until you're ready

Don't autocommit



Optimizing inserts

Commit blocks of data

Local classic or embedded

External tables

Disable indexes



Plans

Show plan

Explicit plans

Alternatives



Making your database slow

Big, comprehensive tables
Skip indexes or
Index everything three different ways
Commit every statement



Summary

Performance starts with design



Questions?



**Envie seu artigo para ser
publicado na
DB FreeMagazine!**

**Valorize-se como profissional
sendo um autor da
DB FreeMagazine.**

Calendário de Eventos

Data	Evento	Site
13/08/2005	BSDDAY São Paulo - SP	http://www.bsdday.com.br/
01 à 05 de Agosto	Encontro Nacional dos Estudantes de Computação Bonito - MS	http://www.enec.org.br/
28/Setembro à 01/Ou- tubro	II Conferência Latino-americana e do Caribe sobre De- senvolvimento e Uso do Software Livre Recife - Olinda - PE	http://twiki.im.ufba.br/bin/view/LacFREE/WebHome
03 à 07 de Outubro	20. Simpósio Brasileiro de Banco de Dados Uberlândia - MG	http://www.sbbd-sbes2005.ufu.br/imagens/ChamadaSBBBD.pdf
3,4,5 Novembro	de Conisli – Congresso Internacional de Software Livre São Paulo - SP	http://www.eventosucesusp.org.br/conisli/

Se você sabe de algum evento focando em **Banco de Dados** ou em **Software Livre** que não esteja listado aqui, envie-nos um email com os dados do evento para que possamos incluí-lo.