



Firebird

O que o torna um banco de dados atraente?

Alexandre Magno FF
:: amagno@argostecnologia.com ::





Agenda

- :: O que é Firebird?
- :: Histórico do banco
- :: Sobre o Firebird Project
- :: Como o Firebird está hoje posicionado no cenário nacional e mundial?
- :: Por que o Firebird é atraente?
- :: Overview da versão atual: Firebird 1.5
- :: O que posso esperar da próxima versão: Firebird 2.0
- :: Estudo de caso – ZCa Madeiras
- :: Firebird Foundation





Pra começar...do que estamos falando?

a) Um mitológico pássaro de fogo



b) Um lendário carro, Pontiac Firebird



c) Um browser do Mozilla.org



d) Um servidor de banco de dados





O que é Firebird?

:: Um verdadeiro SGBDR

:: O Firebird já nasceu grande. Diferente de vários outros bancos de dados open-source, onde muitas vezes não encontramos alguns recursos considerados essenciais em um SGBD relacional, o Firebird conta com: stored procedues, triggers, integridade referencial, SQL (ANSI 92/99), linguagem nativa (PSQL), UDFs (funções definidas por usuários) e outros poderosos recursos.

:: Vale lembrar que a maioria destes recursos foram herdados do InterBase, portanto estão amplamente testados e tem sua eficiência comprovada.





Histórico do banco

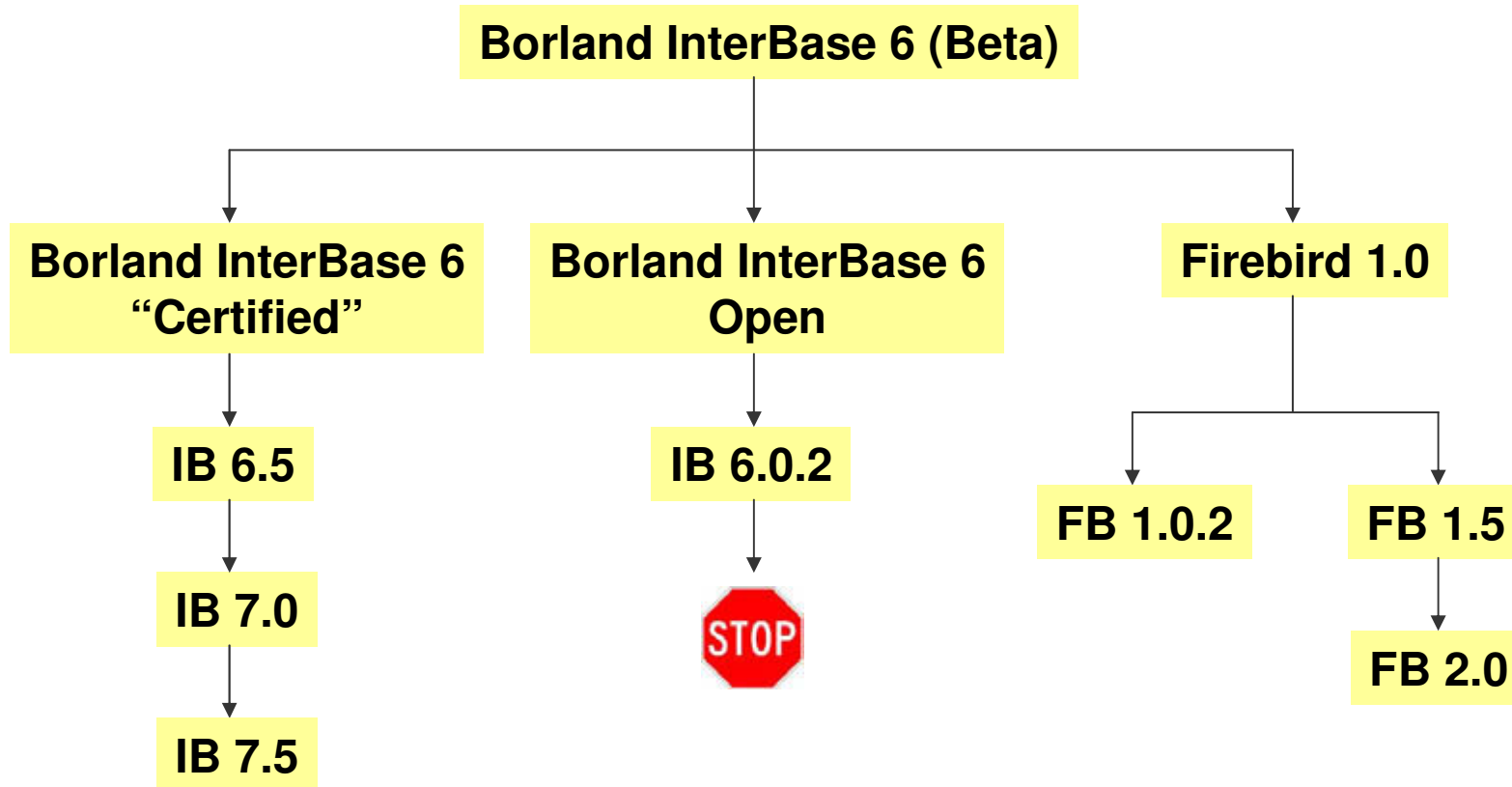
:: Um Roadmap resumido

- 1984 – É lançada a primeira versão do InterBase, ainda chamado Groton Database System
- 1986 – InterBase 2 (já com este nome) é lançado pela Groton
- 1987 – Cognos licencia o código-fonte do InterBase junto a Groton para futuramente lançar seu próprio banco de dados, o StarBase
- 1988 – InterBase 3 é lançado e Groton é adquirida pela Ashton-Tate
- 1991 – Borland adquire Ashton-Tate
- 1992 – Borland lança o InterBase 3.3
- 1994 – Borland lança o InterBase 4
- 1995 – Borland lança Delphi 1.0, com InterBase 4 incluso
- 1995 – Borland lança versões do InterBase para Linux e FreeBSD
- 1996 – InterBase passa a ter suporte a Java através do InterClient, um JDBC Type 3
- 1997 – Borland lança o InterBase 5
- 2000 – Borland lança o InterBase 6 com código-fonte aberto
- 2000 – Nasce o projeto Firebird**





Histórico do banco





Sobre o Firebird Project

:: O projeto!

Como comentado anteriormente, o Firebird nasceu nos bastidores da Borland Software Corporation no momento em que a mesma se pronunciou contra a continuidade do InterBase 6.0 Open. Com isso, algumas das principais figuras relacionadas ao mundo InterBase uniram-se com a comunidade usuária, que estava ansiosa para trabalhar sob o código fonte do InterBase. Algumas pessoas chaves foram:



Jim Starkey
Criador do InterBase



Ann Harrison
Ex-Presidente da InterBase Corp.
(A Borland Company)



Paul Beach
Ex-Sales Manager da InterBase Corp.
(A Borland Company)



*fis*l6.0



Sobre o Firebird Project

:: Algumas estatísticas do projeto

:: Média de 2.500 downloads por semana

:: Média de 15.000 pageviews por dia no SourceForge

:: Build disponível para os principais sistemas operacionais do mercado, para pequeno, médio e grande porte.

:: Vários projetos paralelos ao *core* do banco, tais como de drivers ODBC e JDBC, .NET provider, componentes de acesso, character sets e collates, replicadores, IDE's, *other-databases mode* e outros.

:: Criação de uma fundação para captação de recursos para manutenção do projeto, a Firebird Foundation.





Como o Firebird está posicionado no mercado?

:: Adoção por grandes empresas

:: É cada vez maior a adoção do Firebird por grandes empresas no Brasil e no exterior. Entre as brasileiras podemos citar a Embrapa, Dataprev, Clínica do Leite, Unisal, diversas prefeituras, órgãos públicos e secretarias em todas as regiões e empresas em diversos segmentos.

:: A nível internacional, podemos citar a empresa de telefonia VodaFone que adotou o uso do Firebird em algumas de suas soluções, do pequeno ao grande porte.



:: Há uma lista completa de empresas usuárias do Firebird ao redor do mundo no site do projeto: www.firebirdsql.org



fisl6.0



Como o Firebird está posicionado no mercado?

:: Prêmios recentes

- :: No Brasil ficou em 2º. Lugar na pesquisa da InfoExame de 2004 (Prêmio Info) em solução para banco de dados.
- :: Ficou em 1º. Lugar em pesquisa do EvansData como banco de dados preferido para o desenvolvimento de soluções de workgroup.
- :: Identificado pelo Linux Journal como banco de dados open-source que mais cresceu em número de usuários no ano de 2004.
- :: Possui comunidade no Brasil com dezenas de listas de discussão. A comunidade brasileira de Firebird é considerada uma das maiores do mundo.
- :: Em 2004 foi realizado no Brasil o Firebird Developers Day, que contou com a presença de perto de 400 desenvolvedores.





Mas...o que o torna atraente?

:: Quais características tornam um banco de dados atraente para as empresas?

:: Performance?

:: Alta-disponibilidade?

:: Linguagem poderosa?

:: Facilidade de uso e manutenção?

:: Pequena *footprint*?

:: Popularidade?

:: Multi-plataforma?

:: Suporte a Embedded Applications?

:: Serviços de replicação?





Mas...o que o torna atraente?

:: **Pequeno em tamanho, grande em eficiência**

:: Realmente o Firebird é um banco de dados muito leve no que diz respeito ao tamanho de servidor. Sua instalação completa tem cerca de 4MB.

:: O pequeno tamanho não atrapalha em nada, pelo contrário! O Firebird possui uma alta capacidade de armazenamento e um excelente gerenciamento em projetos com grande volume de tráfego e informação.





Mas...o que o torna atraente?

:: **Simplicidade**

:: A instalação e manutenção de servidor Firebird é extremamente simples. O servidor não requer praticamente qualquer tipo de configuração manual, dispensando na maioria das vezes a necessidade de um DBA.

:: Mesmo assim o Firebird lhe dá recursos para, caso necessário, seja realizado um “fine-tune” no servidor.

:: Os arquivos de banco de dados crescem dinamicamente conforme a necessidade. Não é necessário pré-alocar espaço baseado em deduções de nível de utilização e crescimento dos dados.

:: Todos os objetos do banco (índices, procedures, triggers, etc.) ficam armazenados em um único arquivo, geralmente com a extensão .FDB.





Mas...o que o torna atraente?

:: **Servidor Embutido (Embedded Server)**

:: O servidor embedded é a solução para alguns problemas que sempre acompanharam os desenvolvedores que utilizavam SGBDs relacionais. Com ele, temos um servidor Firebird completo disponibilizado em uma única DLL! É isso mesmo!

:: Ou seja, não é preciso instalar nada, basta distribuir sua aplicação juntamente com o embedded server e não será necessário alterar uma linha de código para que seu programa, que já funciona com o Firebird Server tradicional, funcione também com ele.

:: A única restrição é que apenas uma conexão por banco de dados é permitida (mono-usuário), tornando-o ideal para utilização em catálogos em CD-ROM ou distribuição de versões demo das suas aplicações.





Mas...o que o torna atraente?

:: Multi-plataforma



Mac OS



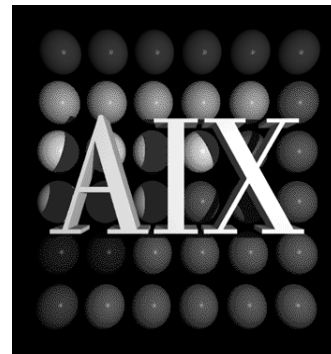
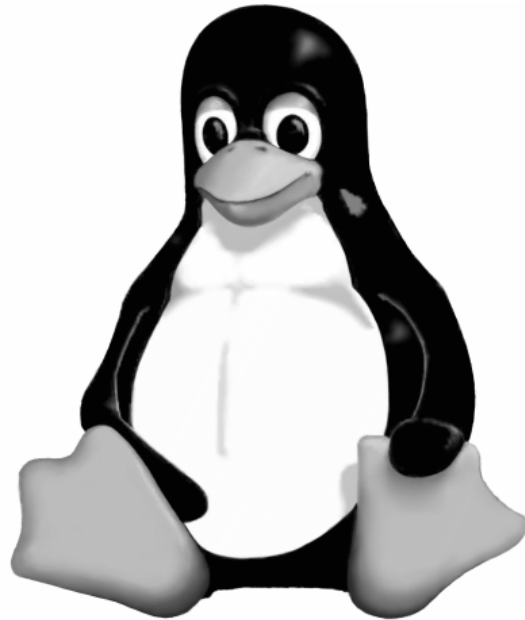
SOLARIS



HP-ux 11i

Version 2

Darwin 



<http://www.freebsd.org>

FreeBSD 

FreeBSD: The Power To Serve



fisl6.0



Mas...o que o torna atraente?

:: **Versioning (MGA)**

:: O Firebird utiliza o versioning para o controle de concorrência no banco de dados. Através deste recurso, múltiplas versões de um mesmo registro podem estar disponíveis no servidor em um determinado momento.

:: Através de um sistema otimista de concorrência, o Firebird permite que transações de leitura não bloqueiem a escrita de outras transações.

:: Tipos de transações disponíveis: ReadCommitted, Concurrency (snapshot) e Consistency

:: Enquanto que em outros bancos de dados o modo de transação snapshot (quando disponível) tem um alto custo para o servidor, no Firebird ele sempre existiu como padrão no versioning.





Mas...o que o torna atraente?

:: **Opções de acesso**

:: Nativo: acesso rápido direto à API do banco. Geralmente é o modo mais eficaz. O acesso pode ser feito diretamente pela API com C/C++ ou via componentes que usam acesso nativo em ferramentas RAD como: IObjects (Delphi/C++Builder), FIBPlus (Delphi/Kylix/C++Builder), etc.

:: JDBC: JayBird Tipo 4 (open-source) para acesso nativo Java.

:: ODBC: Existem inúmeros drivers ODBC para Firebird, alguns pagos, outros free.

:: OLEDB: Para quem usa linguagens da Microsoft, existem vários providers OLEDB para Firebird.

:: .NET Provider: Através do Firebird .NET Provider (open-source)

:: dbExpress (ferramentas Borland)





Mas...o que o torna atraente?

:: **Stored Procedures selecionáveis**

:: Recurso exclusivo do Firebird, permite que uma stored procedure seja utilizada como fonte de dados em um SELECT. Através delas é possível manipular informações dentro do banco de dados, realizando cálculos, condições, etc., e retornar as informações processadas, fazendo o papel de uma tabela virtual.

:: As procedures selecionáveis podem ser utilizadas normalmente com o SELECT padrão, portanto podemos utilizar todos os recursos, como where, order by, etc.

```
SELECT * FROM SP_CLIENTES_BRAZIL('SÃO PAULO', 'ESPECIAL')  
ORDER BY DATA_CADASTRO
```





Mas...o que o torna atraente?

:: UDF – User Defined Functions

:: Através do recurso de criação de UDFs, o Firebird permite com que desenvolvedores não estejam limitados ao conjunto de funções built-in do banco de dados.

:: UDFs são funções criadas em poderosas linguagens de programação e “compiladas” dentro de bibliotecas (dll/so). Você pode escrever UDFs em: C, C++, Delphi, Java, C#, ...)

:: Tais funções (as UDFs) poderão ser utilizadas em qualquer instrução SQL válida para o Firebird, como por exemplo:

```
INSERT INTO CLIENTES(NOME, CPF)
VALUES ('JOHN', VALIDA_CPF(88855577711));
```





Mas...o que o torna atraente?

:: **Alta-disponibilidade com Shadows**

:: Através de shadows, bancos de dados Firebird indisponíveis podem *entrar no ar* rapidamente com pouca ou nenhuma manutenção.

:: Shadows, como o próprio nome diz, são espelhos de bancos de dados on-line, cópias vivas que são atualizadas automaticamente a cada nova tarefa realizada no banco.

:: Com este recurso você consegue proteger seu negócio do risco de indisponibilidade de banco de dados por problemas de estrutura física.

:: Shadows não precisam ser administradas.





Mas...o que o torna atraente?

:: Recursos para replicação

:: A verdade é: qualquer banco de dados que tenha suporte a triggers e mais alguns poucos recursos são capazes de realizar processo de replicação entre bancos de dados.

:: O Firebird facilita esse caminho através do uso de ferramentas (livres ou não) que forneçam um ambiente integrado para configuração e gerenciamento do processo de replicação entre bancos de dados .

:: Replicação one-way, two-way e n-way são suportadas.





Mas...o que o torna atraente?

:: Other-database mode

:: Muito tem se trabalhado em diversos projetos ao redor do mundo para que softwares consigam – a cada dia mais – independência do banco de dados (fornecedores). Mas como fazer isso se:

1. A linguagem ANSI SQL não é poderosa o suficiente para evitar que fornecedores criem sua própria *versão* do SQL.
2. Projetos de software foram (ou ainda são?), durante muito tempo desenvolvidos com a concentração das regras do negócio na camada de banco de dados, ou seja, regras de negócio escritas em uma linguagem proprietária entendida por um único banco de dados.
3. Migrar não é tão simples.



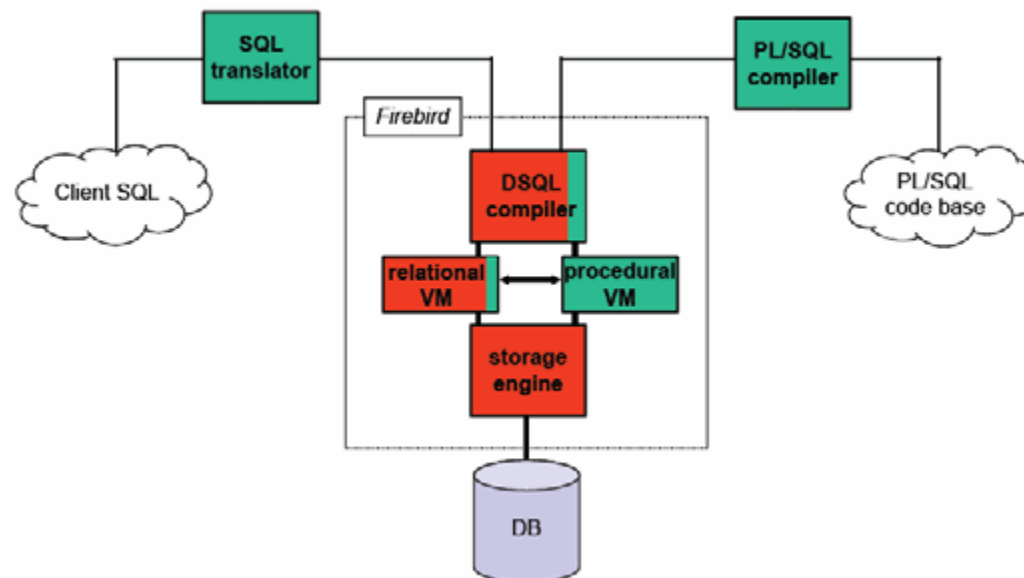


Mas...o que o torna atraente?

:: Other-database mode

:: A tarefa de criar *mode* de servidores de banco de dados é uma forma atraente para compatilizar a utilização de software desenvolvidos para um único servidor de banco de dados com outros servidores.

:: Por exemplo: FYRACLE (Oracle-mode Firebird)





Overview do Firebird 1.5

:: Originalmente esta versão foi planejada para ser um simples porte que suportasse a compilação em C++...só que o Firebird bem além disso!

:: As metas:

- Manter 100% de compatibilidade em interface com o IB 6.x
- Manter mais de 90% de compatibilidade de banco de dados com o IB 6.5 (ou superior).
- Iniciar o processo que permitiria a instalação de ambos, Firebird e InterBase, em um mesmo servidor.
- Incluir novos e poderosos recursos à linguagem.





Overview do Firebird 1.5

:: **Novos recursos de engine**

:: Server-side database aliases

```
Connection String = "Server:AliasName"
```

:: Banco de dados de segurança renomeado para **security.fdb**

:: Nova extensão default: **.fdb**

:: Novo gerenciador de configuração de servidor: **firebird.conf**

:: Nova INTL Interface

:: In-Memory sorting para redução/eliminação de disk I/O





Overview do Firebird 1.5

:: **Novos recursos de engine**

:: Nova implementação do gerenciador de memória

:: Protocolo XNET para builds SuperServer em Win32.

:: Engine Classical para Win32 com suporte mais amigável para SMP





Overview do Firebird 1.5

:: **Novos recursos de SQL**

::suporte a função “CASE”.

:: suporte a função “COALESCE”.

:: suporte a função “NULLIF”.

:: suporte a GROUP BY: por função, sub-query, alias de colunas e ordinal.

:: “SELECT FIRST x” e “ORDER BY” suportado em sub-query.

:: RECREATE VIEW

:: Alias BOOLEAN para um smallint





Overview do Firebird 1.5

:: **Novos recursos de SQL**

:: Lock explícito em instruções SELECT

```
SELECT ... FOR UPDATE WITH LOCK
```

:: Instrução EXECUTE STATEMENT

:: Alias BIGINT para Numeric(18,0)

:: Nome de constraints definidas pelo usuário

:: Suporte a SAVEPOINTS

```
SAVEPOINT savepoint_name;  
ROLLBACK [WORK] TO [SAVEPOINT] savepoint_name;
```





Overview do Firebird 1.5

:: Novos recursos de SQL

:: Triggers universais

```
CREATE TRIGGER mytrigger  
FOR mytable  
BEFORE INSERT OR UPDATE
```

:: Sintaxe CREATE OR ALTER...

:: Novas variáveis de sistema:

```
ROWS_AFFECTED  
TRANSACTION_ID  
CONNECTION_ID  
SQLCODE  
GDSCODE
```





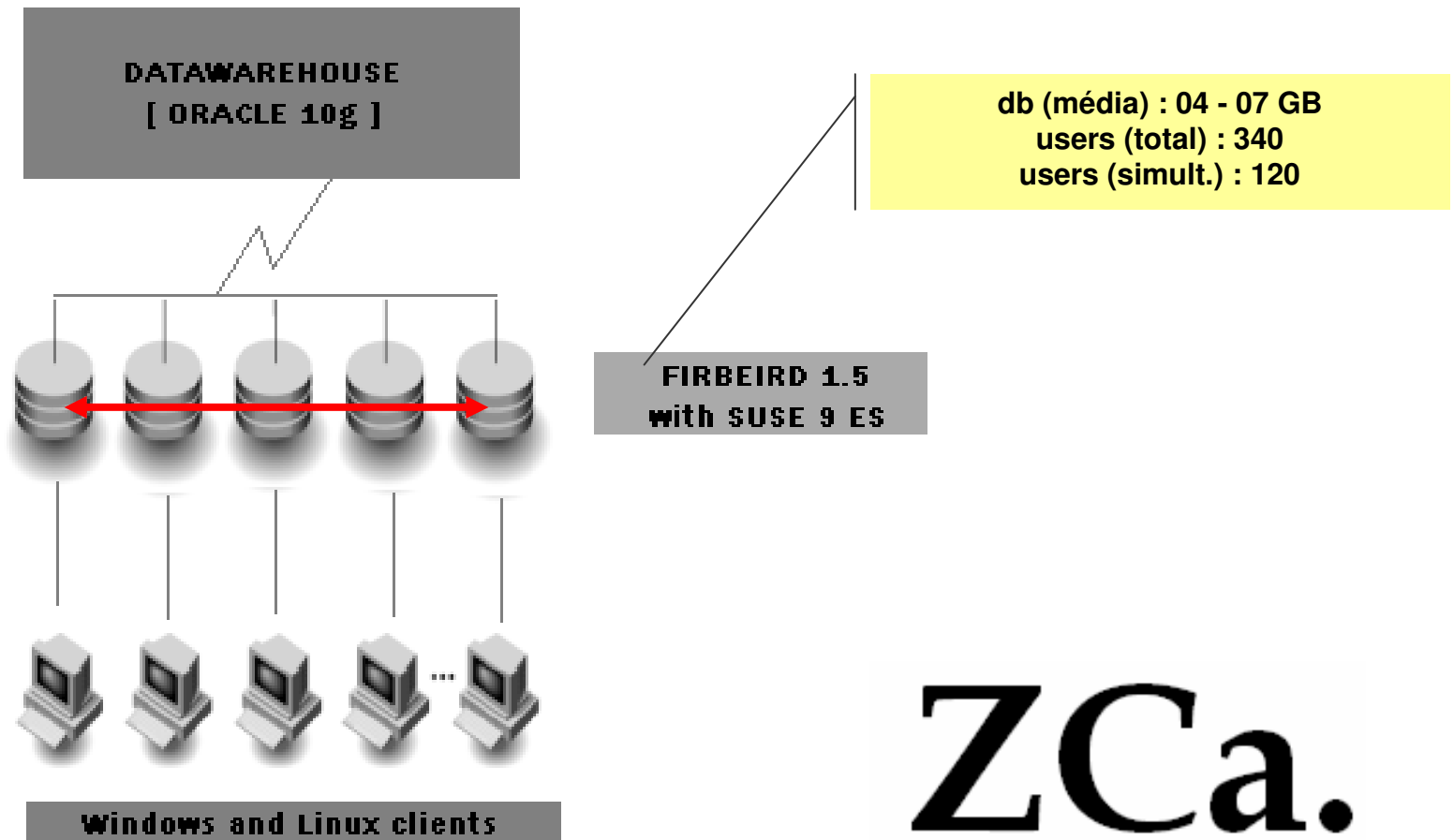
O que posso esperar do Firebird 2.0?

- :: Suporte a tabelas derivadas.
- :: Nova estrutura de índice com um novo layout de estrutura b-tree.
- :: Novas instruções: EXECUTE BLOCK, CANCEL STATEMENT, etc
- :: Nova estrutura de sintaxe para uso de cursores.
- :: Built-in functions.
- :: Novos modos de execução de shutdown de banco de dados: single-user e complete.
- :: **Inovações em SEGURANÇA.**
- :: **Mais performance...mais estabilidade!**





Um pequeno case





Sobre a Firebird Foundation

:: **Objetivos da Firebird Foundation**

- :: Suportar e progredir o desenvolvimento da base do Firebird Project.
- :: Providenciar a infra-estrutura e os mecanismos não comerciais para recolher e gerir os fundos angariados; e distribuir esses mesmos fundos para promover e avançar o esforço de desenvolvimento.
- :: Encorajar a cooperação e a afiliação de indivíduos, outras organizações sem fins lucrativos ou empresas comerciais envolvidas, ou com planos para se envolverem, no desenvolvimento, suporte e promoção de projetos de software baseados em Firebird, assim como em produtos e projetos associados.





LINKS

Firebird Project

www.firebirdsql.org

IBPhoenix Inc.

www.ibphoenix.com

Comunidade Firebird de Língua Portuguesa

www.comunidade-firebird.org

Firebase

www.firebaseio.com.br



fisl6.0



Não perca II Firebird Developers Day

O **Firebird Developers Day** é o maior evento da América Latina dedicado ao Banco de Dados OpenSource FIREBIRD. Na sua primeira edição realizada em 2004, o evento reuniu perto de 400 pessoas, e contou com cerca de 13 horas de informação, divididas em 10 palestras.

O evento esse ano terá a presença especial de dois palestrantes que são ícones internacionais do Firebird: Jim Starkey (*criador do InterBase* e atual desenvolvedor do *Firebird Vulcan*) e Ann Harrison (chefe do Depto. Técnico da *IBPhoenix*).



16 de julho

www.firebirddevelopersday.com.br



fisl6.0



fisl6.0

Obrigado!

Alexandre Magno

amagno@argostecnologia.com

