



Soluções de Conectividade WWW-BD

Helder Rocha

Abril de 1997

Sumário

☞ Parte I - HTML e CGI

- ◆ Breve histórico da World Wide Web
- ◆ Introdução ao HTTP e ao CGI
- ◆ Comunicação com SGBDs
- ◆ Recursos para Oracle e Sybase

☞ Parte II - Java

- ◆ Introdução à linguagem Java
- ◆ Alternativas BD - Java
- ◆ JDBC: Java Database Connectivity
- ◆ Exemplos
- ◆ Produtos



Breve Histórico

➤ A Internet

- rede antiga: mais de 25 anos; rede acadêmica.

➤ Popularidade:

- crescente organização das informações
- podia-se encontrar de tudo na Internet. Aonde?

➤ Gophers... depois WWW.

- mais um serviço: telnet para **info.cern.ch** e logar como “www”.
- palavras seguidas de um número: “links”.
- orientado a caracter
- linguagem HTML
- HTTP: protocolo semelhante ao FTP



Evolução dos browsers

- Popularidade da WWW veio com browsers gráficos.
- Interface colorida e mais tarde interativa
- **O primeiro:** NCSA Mosaic for X-Window; depois Mosaic para Windows
- Netscape
- NS x MSIE



Informações Dinâmicas

- Crescente competitividade da WWW
 - documentos estáticos de texto cada vez menos comuns;
 - meios de interatividade cada vez mais acessíveis e fáceis de usar.
- Interatividade em página web - dois níveis
 - do lado do servidor (CGI)
 - do lado do cliente (Java e JavaScript)
- Cookies



Exemplos

- Yahoo, Altavista, lojas de livros, de CDs, diretórios de e-mail de usuários, sistemas de Internet Banking, sites exclusivos para pagantes, etc.
- Locadoras de vídeo, bibliotecas, revistas, jornais e serviços restritos a assinantes
- Intranet
 - interface multiplataforma,
 - servidor seguro



HTML

- ➔ HTML - Hypertext Markup Language.
 - linguagem de formatação de textos
 - vários elementos usados como descritores
 - define títulos, listas, tabelas e links para textos e imagens
 - descritores entre caracteres especiais “<” e “>”.



HTML

➔ Estrutura básica:

```
<html>
```

```
  <head>
```

```
    <title> Título de uma Página </title>
```

```
  </head>
```

```
  <body>
```

```
    <p> (...)
```

```
  </body>
```

```
</html>
```

➔ Toda a formatação da página depende exclusivamente dos descritores HTML.



CGI

- Common Gateway Interface: permite execução de programas em servidor Web.
- Programas CGI podem ser escritos em qualquer linguagem.
- Métodos HTTP
 - **GET**: linha de comando.
 - **POST**: entrada/saída padrão. Não tem um limite de caracteres.
- Formato de cabeçalho mínimo HTTP:
 - **Content-Type: tipo_de_conteúdo/extensão**
 - Uma linha em branco (fim cabeçalho).
 - Os dados, no formato indicado pelo cabeçalho Content-Type.



Formulários

- Interface HTML para a entrada de dados pelo usuário.
- CGI: processamento do lado do servidor.
- `<FORM>` define um bloco de formulário.
 - ACTION: a URL do programa CGI
 - METHOD, o método do protocolo HTTP
- Objetos do formulário
 - INPUT, SELECT, TEXTAREA
 - ◆ NAME identifica uma variável
 - ◆ VALUE identifica valor armazenado.



Formulários

- ➔ URL-encoding. Para decodificar:
 - 1. Converta as seqüências hexadecimais %hh para os caracteres equivalentes (ASCII)
 - 2. Converta os sinais de “+” em espaços.
 - 3. Separe os pares nome/valor pelo “&”
 - 4. Identifique o nome e o valor separados por “=”.



Algumas aplicações de CGI

- Contador gráfico
- Programa de Busca
- FormMail
- Gateways



Gateways CGI para SGBDs

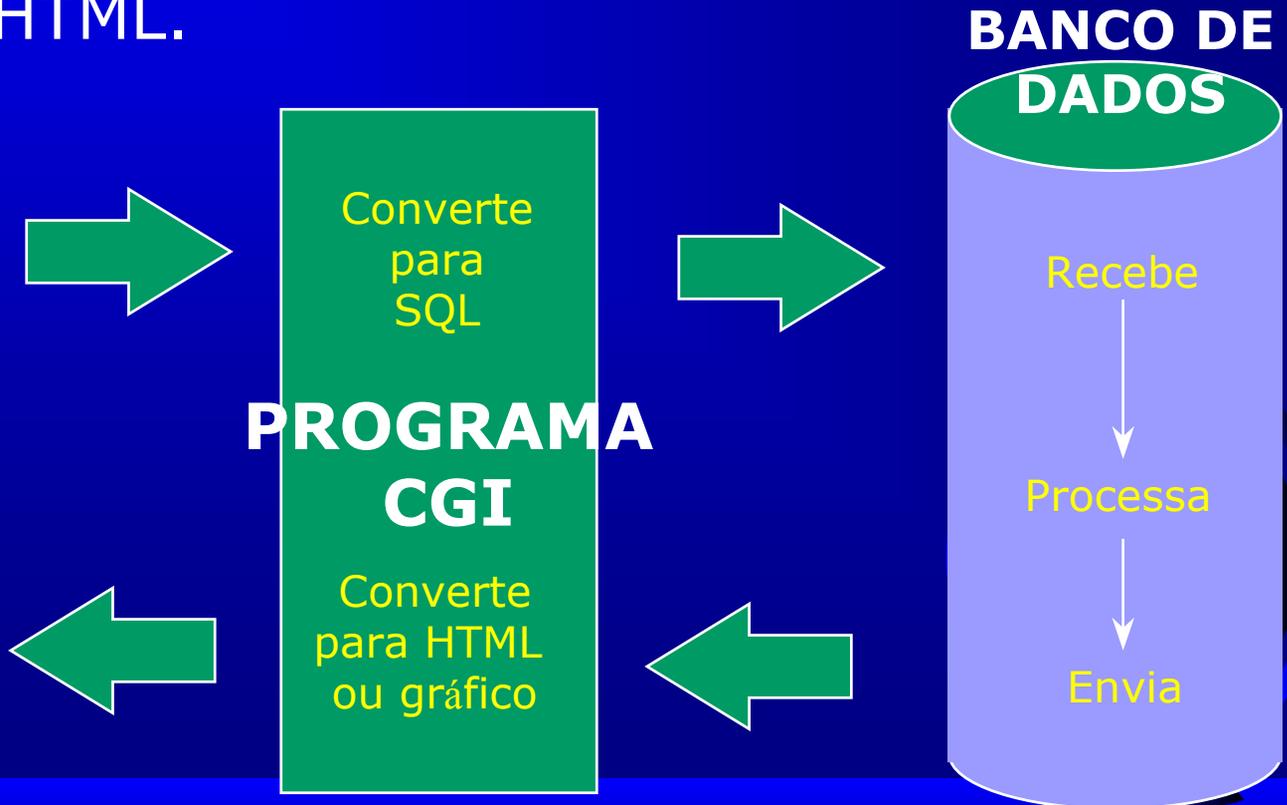
- ➔ Converte comandos de formulário HTML para a linguagem sistema de banco de dados.
- ➔ Converte respostas do sistema de banco de dados p/ HTML.



Formulário



Resultado



Exemplo: tabela

Livro			
Título	Autor	Editora	ISBN
Sphere	Michael Crichton	Ballantine	0-345-35314-5
Mr. Vertigo	Paul Auster	Faber and Faber	0-571-17345-4
The Lost World	Michael Crichton	Ballantine	0-345-40507-2
A Nova Califórnia	Lima Barreto	Círculo do Livro	85-332-0524-4
Cronica de una Muerte Anunciada	Gabriel Garcia Marquez	Plaza & Janes	84-01-42270-01
Interview with the Vampire	Anne Rice	Ballantine	0-345-33766-2
Livro de Sonetos	Vinícius de Moraes	Círculo do Livro	85-332-0139-5



Exemplo: formulário

Form - Microsoft Internet Explorer

File Edit View Go Favorites Help

Address E:\JavaScript\bibliosrch.html Links 

[DIRETÓRIOS](#) | [COLEÇÕES](#) | [REVISTAS](#) | [DISSERTAÇÕES](#) | [AQUISIÇÕES](#) | [BOLETIM](#) | [HOME PAGE](#)

Pesquisa Livros na Biblioteca

Neste formulário você pode procurar um livro pelo Autor, Título, Editora ou ISBN. Você pode fornecer informações parciais (ex: sobre nome do autor, uma palavra do título) desde que sejam palavras inteiras e preencher quantos campos quiser (pelo menos um, se você quiser achar alguma coisa).

Autor:

Título:

Editora:

ISBN:

Dica: Faça uma busca geral primeiro. Se a listagem for muito grande, faça uma busca mais fechada, especificando mais detalhes sobre a obra que procura.

Se você conhece um livro que deveria estar aqui e não está, sugira a sua compra ao departamento através da página [Aquisições](#).



Exemplo: SQL

- Campo do formulário:
 - **nome** de variável armazena **valor** digitado pelo usuário.

- Monta instrução para BD:

```
select *  
  from Livro  
  where Autor = 'Michael Crichton'
```

- Resposta do CGI



Aplicações

- Interface mais completa: leitura, alteração, escrita
- Locadora:
 - busca para encontrar o filme
 - locação
 - alteração do registro para indicar locação



Gráficos

- O programa CGI pode chamar outros programas
- Aplicações: gráficos.
- CGI passa para aplicativo de construção de imagens valores necessários
- É necessário informar tipo de conteúdo:
Content-Type
 - Content-Type: image/gif
 - Content-Type: image/jpg



Aplicações Oracle e Sybase

- **OraPerl:** subrotinas em Perl para gerar pesquisas em um formato reconhecível pelo banco de dados Oracle.

<http://src.doc.ic.ac.uk/packages/perl/db/perl4/oraperl>

- **SybPerl:** subrotinas em Perl para gerar pesquisas em um formato reconhecível pelo banco de dados Sybase.

<http://src.doc.ic.ac.uk/packages/perl/db/perl4/sybperl>



Questões de Segurança: SSL

- Não se pode confiar no meio por onde são enviados os comandos.
- Qualquer computador ou roteador no meio do caminho pode interceptar os dados (inclusive recuperar senhas e códigos de acesso).
- Solução: criptografia: servidor seguro com SSL ou SHTTP



O que é Java?

- Linguagem orientada a objetos desenvolvida pela Sun
- Muito popular na Internet e fora dela
- Muito parecida com C/C++. É um “C” mais simples e mais poderoso.
- É independente de plataforma, segura, robusta, *multi-threaded*, eficiente e de alto desempenho.



O que é Java (2)

- Tudo em Java são classes e objetos
- Os métodos (funções) e variáveis sempre pertencem ou a uma classe ou a um objeto.
- Classes são organizadas em “pacotes”
- Java não suporta herança múltipla de classes
- Suporta “herança” de múltiplas “interfaces”
 - “interfaces” são classes especiais cujos métodos são apenas declarados



O que é Java (3)

- Há dois tipos básicos de aplicações Java:
 - applets: aplicação que executa dentro de um browser (é limitada pela capacidade do browser)
 - aplicação *standalone*: aplicação que roda independente de browser (como qualquer outra aplicação desenvolvida em outra linguagem)
- Aplicações Java, depois de compiladas, rodam em qualquer lugar.



Java X C/C++

- Java é bem mais simples que C/C++
- Não tem apontadores (não tem acesso direto a posições de memória)
- Tem um sistema de coleta de lixo, que dispensa liberação de memória
- Usa exceções (como Delphi)
- É uma evolução do C para orientação a objetos, sem a complexidade de C++



Applets

- ☞ Applets são aplicações que rodam em um browser.
- ☞ Tem grande potencial na Internet e Intranet
- ☞ É a aplicação da Internet e do Network Computer
- ☞ Applets remotos passam por uma verificação local --> Segurança



Applets versus aplicações

☞ Applets

- não rodam de forma independente
- dependem de aplicativo gráfico (geralmente um browser)
- são sempre programas gráficos
- execução depende de chamada através de uma página HTML.

☞ Aplicações

- rodam independentemente
- podem ser gráficas, podem ser de linha de comando e podem sequer aparecer na tela.



API, classes e pacotes

- API Java: biblioteca de todas as classes e interfaces pré-definidas na linguagem.
- Organização em “pacotes”.
- Cada pacote é um conjunto de classes que tem alguma função comum.
- Pacotes da versão 1.02:
 - `java.lang`, `java.awt`, `java.awt.image` e `java.awt.peer`, `java.io`, `java.net`, `java.util`, `java.applet`
- Identificação universal = `pacote.classe`
 - `java.lang.System`
- Classes precisam ser importadas antes de usadas



Segurança das applets

- Identificar applets seguros através de um certificado (disponível na versão 1.1).
- Garantir que um determinado applet é confiável para ampliar as suas capacidades.



Java: Estado Atual

- Versão 1.0 de Java lançada em Jan/96
- Versão atual é 1.1 (mais popular ainda é 1.02)
- Distribuição gratuita através do JDK (Java Development Kit) na JavaSoft:
 - <http://www.javasoft.com>
 - (ou <http://java.sun.com>)
- Contém todas as classes e interfaces necessárias para se trabalhar com bancos de dados



Java é ideal para BDs

☞ segura

- garante integridade do BD

☞ produtividade do desenvolvimento maior

- verificação mais rígida da compilação, gerenciamento de memória otimizado

☞ flexibilidade adicional

- carregamento dinâmico, métodos nativos, tipos, cache de dados no cliente

☞ familiar

- sintaxe muito parecida com C/C++



Alternativas de conectividade a BDs com Java

- Bancos de Dados Relacionais
 - JDBC (Java Database Connectivity)
- Bancos de Dados Orientados a Objetos
 - ODMG, capítulo 7
- Integração BD Relacional-BD Objeto
- Serialização de objetos: persistência
 - API RMI/Serialization



Serialização

- Permite implementação de objetos “persistentes”
- Parte da API Java RMI - Remote Method Invocation (JDK 1.1)
- Permite que um objeto Java possa ser salvo de forma persistente
 - é necessário fazer chamadas explícitas para serializar cada objeto
 - existem formas menos rudimentares com JDBC e ODMG



Bancos de Dados de Objetos

- Usam modelo de objetos (não relacional)
- Persistência transparente para objetos
- Há uma API padrão para Java sendo desenvolvida pelo ODMG (Object Database Standards Consortium)
- Já existem produtos para BDs de Objetos
 - Object Design (<http://www.odi.com>)
 - POET (<http://www.poet.com>)



Java em BDs Relacionais

- Desde que Java foi lançada, há grande interesse em desenvolver uma API para acesso a BDs relacionais
- Vantagens de Java levaram empresas a desenvolver APIs proprietárias para este fim
- Para evitar uma “torre de Babel” de implementações diferentes no futuro, a JavaSoft desenvolveu o JDBC



O que é JDBC?

- **Java Database Connectivity:** uma interface padrão de acesso a bancos de dados SQL oferecendo acesso uniforme a uma gama de bancos de dados relacionais.
- Um programa Java pode enviar instruções SQL para qualquer banco de dados SQL.
- Pode-se escrever um único programa que será capaz de enviar instruções SQL para o banco de dados correto.



JDBC:

Java Database Connectivity

- É uma API independente de SGBD
- Define classes Java para representar
 - conexões de rede
 - declarações SQL
 - conjuntos de resultados
 - conjuntos de dados
- Baseado no padrão X/Open Call Level Interface (CLI)
- Espelhado no ODBC

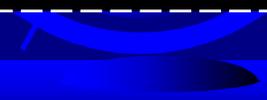
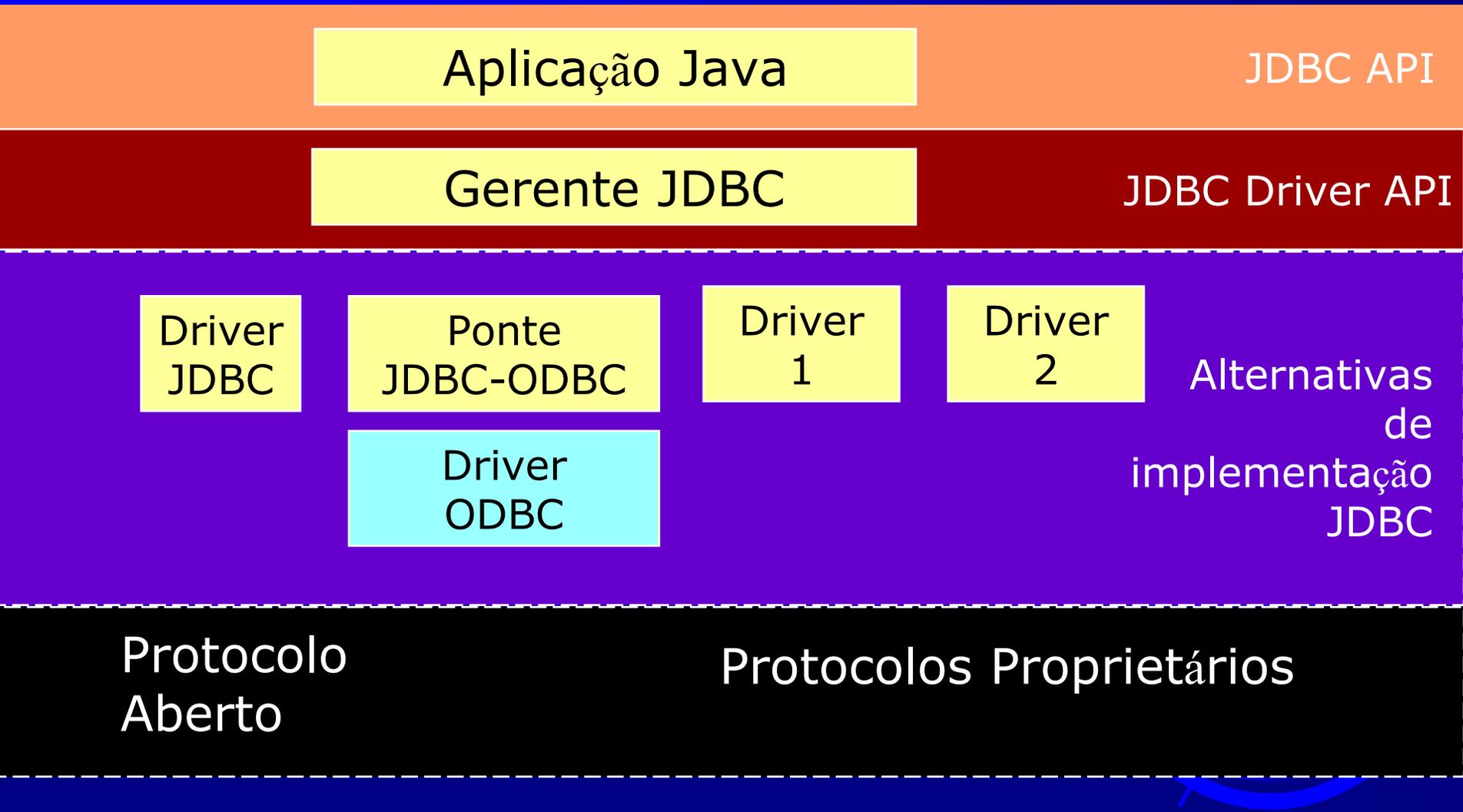


JDBC: Java Database Connectivity

- Através de um “Gerente de Drivers”, drivers proprietários podem ser usados para conectar a diversos SGBDs
- Driver especial: “Ponte ODBC-JDBC”
 - suporta JDBC através de drivers ODBC
 - melhor forma de usar ODBC através de Java



Diagrama de Camadas JDBC



Classes da API JDBC

- A versão 1.1 consiste de 16 classes, interfaces e exceções
- As principais (interfaces) são:
 - `java.sql.DriverManager` (classe)
 - `java.sql.Connection`
 - `java.sql.Statement`
 - `java.sql.ResultSet`
 - `java.sql.Driver`



O que JDBC faz?

- Em resumo, JDBC dá condições de realizar três operações:
 - estabelecer uma conexão com um banco de dados
 - enviar instruções SQL
 - recuperar e processar os resultados.
- JDBC é uma interface de baixo nível. Interfaces mais amigáveis podem ser construídas em cima dela.



Exemplo

```
Connection con =  
    DriverManager.getConnection("jdbc:odbc:rigel",  
        "login", "passwd");  
  
Statement stmt = con.createStatement();  
  
ResultSet rs = stmt.executeQuery ("SELECT  
    Número, Título, Preço FROM Livro");  
  
    while (rs.next()) {  
        int x = getInt("Número");  
        String s = getString("Autor");  
        float f = getFloat("Preço");  
    }  
}
```



Produto JDBC da Sun

- O produto JDBC consiste de três componentes:
 - O JDBC driver manager,
 - JDBC driver test suite e
 - ponte JDBC-ODBC.
- Ponte JDBC-ODBC permite que drivers ODBC possam ser usados como drivers JDBC.
- Forma de acessar SGBDs menos populares que ainda não tem um driver JDBC



URL JDBC

- A URL usada no JDBC tem a sintaxe seguinte:
 - ◆ `jdbc:subprotocolo:subnome`
- O subprotocolo é o nome do tipo de protocolo de banco de dados que está sendo usado para interpretar o SQL.
- O subnome é o nome que o subprotocolo utilizará para localizar um determinado servidor ou base de dados. Exemplos:
 - ◆ `jdbc:odbc:servidor1`
 - ◆ `jdbc:odbc:servidor1:80/web/data`



Como usar JDBC - Exemplo de uso: Recuperando informação

- Exemplo comentado do tutorial da Sun (veja material impresso)
- Programa em Java que pode ser usado como gabarito de teste (basta substituir a URL, login, senha e incluir a instrução SQL que você deseja enviar para o banco de dados).
- Consulte o tutorial da Sun para a listagem original completa.



Produtos: Categorias de Drivers JDBC disponíveis

- Ponte JDBC-ODBC + driver ODBC
- Native-API partly-Java driver
- JDBC-Net pure Java driver
- Native-protocol pure Java driver
 - Veja <http://www.javasoft.com/products/jdbc> para dados mais atualizados.
 - Os primeiros fabricantes com drivers na categoria 3 foram SCO, Open Horizon, Visigenic e WebLogic.



Drivers JDBC

- Cat. 1: Ponte JDBC-ODBC
 - JavaSoft (suporta vários SGBDs via ODBC)
- Cat. 2: Driver API-nativa parcialmente-Java
 - Intersolv DataDirect (Oracle, Sybase), IBM DB2
- Cat. 3: Driver protocolo de rede “pure Java”
 - Asgard, DataRamp, Intersoft Essentia, Openlink, SCO, Symantec, Visigenic, WebLogic
- Cat. 4: Driver protocolo nativo “pure Java”
 - Imaginary mSQL-JDBC, SAS Share-Net, Borland Interclient, Connect (InterBase 4, Sybase, Oracle, Informix, Ingres, ADABAS)



Aplicações

- Applets podem usar JDBC para acesso a Bancos de Dados via Internet ou na Intranet
- Restrições de segurança dos Applets podem ser flexibilizadas através de certificados para ter mais acesso.
- Aplicações Java C/S rodando em uma Intranet podem acessar diferentes SGBDs em diferentes plataformas



Tendências

☞ Java Object-Relational Mapping

- Sun planeja desenvolver uma nova API sobre JDBC, para mapear objetos Java transparentemente para linhas de uma ou mais tabelas relacionais

☞ Java Transaction Service

- permite que um programa Java realize uma transação distribuída através de múltiplos BDs



Mais Informação, Especificações, Código, API, etc...

☞ JavaSoft

– <http://java.sun.com>

☞ JavaSoft JDBC Home Page

– <http://splash.javasoft.com/jdbc>

☞ ODMG Java Page

– <http://www.odmg.org/java.html>

☞ Object Serialization

– <http://chatsubo.javasoft.com>

