

IBM DB2 Alphablox



Introdução

Versão 84

IBM DB2 Alphablox



Introdução

Versão 84

Nota:

Antes de utilizar estas informações e o produto suportado por elas, leia as informações nos “Avisos” na página 47.

Terceira Edição (Setembro de 2006)

Esta edição aplica-se à versão 8, release 4, do IBM DB2 Alphablox para Linux, UNIX e Windows (produto número 5724-L14) e a todos os releases e modificações subseqüentes, até que seja indicado de outra forma em novas edições.

Quando o Cliente envia seus comentários, concede direitos não-exclusivos à IBM para usá-los ou distribuí-los da maneira que achar conveniente, sem que isso implique em qualquer compromisso ou obrigação para com o Cliente.

Direitos Autorais © 1996 - 2006 Alphablox Corporation. Todos os direitos reservados.

© Direitos Autorais International Business Machines Corporation 1996, 2006. Todos os direitos reservados.

Índice

Capítulo 1. Instalando Bancos de Dados de Amostra no Alphablox 1

Construindo um Cubo a partir de uma Origem de Dados Relacionais de Amostra	2
Instalando os Dados de Amostra	2
Definindo Origens de Dados	6
Criando a Definição de Cubo	7
Instalando Dados de Amostra Multidimensionais	8
Instalação de Dados de Amostra para DB2 OLAP Server V8.1 e Hyperion Essbase V6.5	8
Instalação de Dados de Amostra para DB2 OLAP Server V8.2 e Hyperion Essbase V7.1	9
Instalando Dados de Amostra do Microsoft SQL Server 2000 Analysis Services	11
Instalando Dados de Amostra do Microsoft SQL Server 2005 Analysis Services	12
Definindo a Origem de Dados	13

Capítulo 2. Tutorial: Construindo Seu primeiro Aplicativo 15

Definindo seu Aplicativo	15
Acessando Dados	16
Criando a Página Inicial do Aplicativo	16
Configurando uma Página Inicial Padrão	17
Criando a Primeira Visualização Analítica	18
Estrutura da Página de sua Primeira Visualização Analítica	21
Criando uma Visualização Analítica Customizada	23
Resumo	24

Capítulo 3. Tutorial: Construindo seu Primeiro Portlet com Componentes Blox 25

Instalando Portlets de Amostra	25
Executando os Portlets de Amostra	26
Examinando a Estrutura de uma Página JSP do Portlet com um Componente Blox	26
Criando sua Própria Página JSP do Portlet com Componentes Blox	28
Criando um Projeto de Portlet para Utilizar Componentes Blox	30

Configurando um Projeto de Portlet Utilizando o Rational Application Developer.	31
Próximas Etapas.	32
Dicas de Desenvolvimento de Portlets	33

Capítulo 4. Tutorial: Construindo Aplicativos Utilizando Ferramentas Rational Developer 35

Preparando seu Ambiente de Desenvolvimento	35
Instalando o DB2 Alphablox Toolkit	35
Instalando o DB2 Alphablox nos Ambientes de Teste Integrados do WebSphere.	36
Criando uma Instância do Servidor WebSphere	37
Criando Variáveis de Substituição do Servidor WebSphere 5.1	37
Configurando as Instâncias do Servidor WebSphere 5.1	38
Incluindo o Usuário Guest no Grupo de Administradores do DB2 Alphablox	39
Criando Aplicativos do DB2 Alphablox	39
Criando Arquivos JSP com o Conteúdo do DB2 Alphablox	39
Acessando Origens de Dados do DB2 OLAP Server e do Essbase	40

Capítulo 5. Tutorial: Construindo um Cubo do DB2 Alphablox, Utilizando o DB2 Cube Views 43

Definindo uma Origem de Dados Relacionais do DB2	43
Definindo uma Origem de Dados do Alphablox Cube Server Adaptor	44
Definindo um Cubo do DB2 Alphablox	45
Iniciando o Cubo do seu DB2 Alphablox	46

Avisos 47

Marcas Registradas	49
------------------------------	----

Índice Remissivo 51

Capítulo 1. Instalando Bancos de Dados de Amostra no Alphablox

Dados de amostra da QCC (Quality Chocolate Company) são utilizados nos tutoriais do DB2 Alphablox, nos exemplos de aplicativos do Blox Sampler e em vários dos códigos de amostra do DB2 Alphablox Information Center.

Estes dados de amostra estão disponíveis para diferentes bancos de dados:

Relacional

Você deve instalar uma das versões relacionais de dados de amostra da QCC para acompanhar o primeiro tutorial para construir seu primeiro aplicativo. O cubo da QCC criado a partir dos dados relacionais é utilizado em tutoriais e nos exemplos de aplicativos do Blox Sampler.

- DB2
- Apache Derby
- Informix
- Microsoft SQL Server
- Oracle
- Sybase

Multidimensionais

O conjunto de exemplos do Blox Sampler também é executado nos seguintes bancos de dados multidimensionais. Eles não são obrigatórios para tutoriais. No entanto, se você estiver utilizando um dos bancos de dados a seguir em seu ambiente, deverá instalar os dados de amostra da QCC para tirar proveito dos exemplos reais nos aplicativos do Blox Sampler.

- IBM DB2 OLAP Server V8.1; Hyperion Essbase V6.5
- IBM DB2 OLAP Server V8.2; Hyperion Essbase V7.1
- Microsoft SQL Server 2000 Analysis Services
- Microsoft SQL Server 2005 Analysis Services

Informações sobre a QCC

Nossa empresa hipotética, a Quality Chocolate Company, é uma cadeia de lojas de varejo em rápido crescimento que vende uma pequena seleção de confeitos tradicionais de gourmet, além de algumas especialidades sazonais. Em poucos anos a QCC, que começou em uma cidade pequena, ampliou tanto o tamanho de sua linha de produto quanto o número de localidades. Agora a empresa gostaria de realizar análise de vendas, análise de margem bruta e análise de orçamento. Além disso, a QCC também deseja controlar os vários atributos de seus produtos, como o tipo de chocolate (meio amargo, ao leite ou puro) e se seus produtos contém nozes, para ajudar na determinação e análise de preferências dos clientes.

Construindo um Cubo a partir de uma Origem de Dados Relacionais de Amostra

São fornecidas tabelas relacionais para os dados de amostra da QCC (Quality Chocolate Company) e um arquivo de propriedades de definição para que você possa criar um cubo Alphablox a partir de seus dados relacionais.

A construção de um cubo Alphablox a partir dos dados relacionais da QCC fornecidos envolve três etapas gerais:

1. Instalar as tabelas de amostra.
2. Definir as origens de dados da origem de dados relacional subjacente e do cubo que será construído na origem de dados relacionais através das Páginas de Administração do DB2 Alphablox.
3. Criar a definição de cubo armazenando o arquivo de propriedades de definição do cubo em seu Repositório do DB2 Alphablox. Esse arquivo de propriedades contém as informações que descrevem as dimensões, medidas e níveis do cubo e onde os dados são armazenados na origem de dados relacionais subjacente.

Instalando os Dados de Amostra

Para que seja possível construir um cubo Alphablox com os dados de amostra da QCC (Quality Chocolate Company) é necessário instalar as tabelas da QCC em seu servidor de banco de dados.

Selecione as instruções de instalação apropriadas para o servidor de banco de dados.

Instalando os Dados de Amostra do DB2

Dados de amostra são fornecidos para origens de dados do DB2. É necessário instalar as tabelas de amostra para que seja possível utilizá-las para construir um cubo Alphablox requerido para o primeiro tutorial.

Para instalar os dados de amostra da QCC (Quality Chocolate Company) para origens de dados do DB2:

1. Copie todos os arquivos do diretório `sampledata/qcc/db2` do CD de instalação do Alphablox em um diretório na máquina que possui o cliente DB2 instalado.
2. Inclua sua conexão do DB2 no Control Center.
3. A partir de um prompt de comandos, vá para o diretório onde copiou os arquivos e execute o arquivo `qcc_db2.bat` com os parâmetros de seu nome de banco de dados, nome de usuário e senha do DB2. Por exemplo: `qcc_db2 test db2admin password`. Este comando efetua login em seu banco de dados DB2 denominado `test`, como o usuário `db2admin` com a senha `password` e carrega os dados na tabela `qcc`. Dois arquivos são criados:
 - Um arquivo `db2_command.txt` que inclui o parâmetro de conexão do DB2 e o arquivo `db2_qcc.sql`.
 - Um arquivo de registro `qcc.log` para carregar os dados da QCC.

Nota: A tabela `qcc` será eliminada se ainda existir.

4. Visualize as mensagens do arquivo de registro.

As tabelas são instaladas agora.

Continue com a seção “Definindo Origens de Dados” na página 6 para definir tanto a origem de dados relacional quanto a origem de dados do cubo para o DB2 Alphablox.

Instalando os Dados de Amostra para Derby

Dados de amostra são fornecidos para origens de dados Derby. É necessário instalar as tabelas de amostra para que seja possível utilizá-las para construir um cubo Alphablox requerido para o primeiro tutorial.

Para instalar os dados de amostra da QCC (Quality Chocolate Company) para origens de dados Derby:

1. Copie todos os arquivos do diretório `sampledata/qcc/derby` do CD de instalação do Alphablox em um diretório na máquina que possui o servidor Derby instalado.
2. Abra o arquivo `qcc_dby.bat` em um editor de texto.
3. Verifique os caminhos `DERBY_INSTALL` e `JAVA_INSTALL` e os altere se for necessário. O valor padrão para `DERBY_INSTALL` é `C:\Arquivos de programas\IBM\Cloudscape_10.0` e o padrão para `JAVA_INSTALL` é `C:\Arquivos de programas\IBM\Cloudscape_10.0\ibm-jre-n142p\jre\bin`.
4. A partir de um prompt de comandos, vá para o diretório onde copiou os arquivos e execute o arquivo `qcc_dby.bat` com parâmetros de seu nome de banco de dados, nome de usuário e senha do DB2. Isso cria um banco de dados e tabelas do Blox com o usuário `abx`. Um arquivo de registro `qcc.log` também é criado para carregar os dados da qcc.

Nota: A tabela QCC será eliminada se ainda existir.

5. Visualize as mensagens do arquivo de registro.

As tabelas são instaladas agora.

Continue com a seção “Definindo Origens de Dados” na página 6 para definir tanto a origem de dados relacional quanto a origem de dados do cubo para o DB2 Alphablox.

Instalando os Dados de Amostra do Informix

Dados de amostra são fornecidos para origens de dados do Informix. É necessário instalar as tabelas de amostra para que seja possível utilizá-las para construir um cubo Alphablox requerido para o primeiro tutorial.

Certifique-se de ter criado as variáveis de ambiente `INFORMIXDIR` e `INFORMIXSERVER`. Consulte a publicação IBM Informix *DB-Access User's Manual* para obter informações adicionais sobre o uso do **DB-Access**.

Para instalar os dados de amostra da QCC em origens de dados Informix:

1. Crie um banco de dados onde deseja carregar os dados de amostra em seu servidor Informix.
2. Copie todos os arquivos do diretório `sampledata/qcc/informix` no CD de instalação do Alphablox para a máquina onde o servidor Derby está instalado.
3. A partir de um prompt de comandos, vá para o diretório onde salvou os arquivos e digite `dbaccess`.
4. Conecte-se ao banco de dados onde deseja carregar os dados de amostra.
 - a. Selecione **DBAccess** → **Conexão**.
 - b. Selecione **Conectar**.

- c. Selecione o servidor de banco de dados.
 - d. Digite o nome de usuário do administrador de banco de dados.
 - e. Digite a senha.
 - f. Selecione o banco de dados. O banco de dados está agora conectado.
 - g. Selecione **Sair**.
5. Carregue os arquivos SQL para criar tabelas.
 - a. Selecione **DBAccess** → **Query-language**.
 - b. Selecione **Escolher**.
 - c. Selecione o arquivo `qcc_inf.sql`.
 - d. Selecione **executar**. As tabelas são criadas.
 - e. Selecione **Sair** para retornar para o prompt de comandos.
 6. Carregue os dados. Execute o arquivo `qcc_inf.bat` a partir de um prompt de comandos com o parâmetro do nome do banco de dados. Por exemplo:
`qcc_info blox`.

Nota: Um arquivo de registro `qcc.log` é criado após a execução do arquivo em lote.

As tabelas são instaladas agora.

Continue com a seção “Definindo Origens de Dados” na página 6 para definir tanto a origem de dados relacional quanto a origem de dados do cubo para o DB2 Alphablox.

Instalando os Dados de Amostra da Microsoft

Dados de amostra são fornecidos para origens de dados do Microsoft SQL Server. É necessário instalar as tabelas de amostra para que seja possível utilizá-las para construir um cubo Alphablox requerido para o primeiro tutorial.

Para instalar os dados de amostra da QCC (Quality Chocolate Company) para origens de dados do Microsoft SQL Server:

1. Copie todos os arquivos do diretório `sampledata/qcc/mssql` do CD de instalação do Alphablox para um diretório de sua máquina.
2. A partir de um prompt de comandos, execute o arquivo `qcc_sql.bat` e especifique os parâmetros de conexão de seu servidor SQL (*server database table owner user password*). Por exemplo, `qcc_sql server12 blox dbo sa`. Esse comando executa o arquivo em lote e conecta ao banco de dados do servidor SQL no servidor `server12`, com o banco de dados denominado `blox`, proprietário de tabela `dbo`, nome de usuário `sa` e senha `null`. Executar o script `qcc_sql.bat` cria os seguintes arquivos de registro:
 - Um arquivo `qcc.log` para criação de tabelas da `qcc`.
 - Um arquivo `qcc_fact.log` para carregamento de dados na tabela `qcc_fact`.
 - Um arquivo `qcc_locations.log` para carregamento de dados na tabela `qcc_locations`.
 - Um arquivo `qcc_products.log` para carregamento de dados na tabela `qcc_products`.
 - Um arquivo `qcc_scenarios.log` para carregamento de dados na tabela `qcc_scenarios`.
 - Um arquivo `qcc_time.log` para carregamento de dados na tabela `qcc_time`.

Nota: A tabela `qcc` será eliminada se ainda existir.

3. Consulte quaisquer mensagens nos arquivos de registro.

As tabelas são instaladas agora.

Continue com a seção “Definindo Origens de Dados” na página 6 para definir tanto a origem de dados relacional quanto a origem de dados do cubo para o DB2 Alphablox.

Instalando os Dados de Amostra para Oracle

Dados de amostra são fornecidos para origens de dados Oracle. É necessário instalar as tabelas de amostra para que seja possível utilizá-las para construir um cubo Alphablox requerido para o primeiro tutorial.

Para instalar os dados de amostra da QCC (Quality Chocolate Company) para origens de dados Oracle:

1. Copie todos os arquivos do diretório `sampledata/qcc/oracle` no CD de instalação do Alphablox para a máquina onde o cliente Oracle está instalado.
2. Execute o arquivo `qcc_ora.bat` com os parâmetros de sua conexão Oracle. Por exemplo, no prompt do DOS, digite `qcc_ora server1 oraadmin tiger`. Esse comando executa o arquivo em lote e conecta ao banco de dados `server1` do Oracle com o nome de usuário `oraadmin` e senha `tiger`. Os arquivos de registro a seguir são criados quando você executa o arquivo em lote:
 - Um arquivo `qcc.log` para criação de tabelas da qcc.
 - Um arquivo `qcc_fact.log` para carregamento de dados na tabela `qcc_fact`.
 - Um arquivo `qcc_locations.log` para carregamento de dados na tabela `qcc_locations`.
 - Um arquivo `qcc_products.log` para carregamento de dados na tabela `qcc_products`.
 - Um arquivo `qcc_scenarios.log` para carregamento de dados na tabela `qcc_scenarios`.
 - Um arquivo `qcc_time.log` para carregamento de dados na tabela `qcc_time`.
3. Consulte quaisquer mensagens nos arquivos de registro.

As tabelas são instaladas agora.

Continue com a seção “Definindo Origens de Dados” na página 6 para definir tanto a origem de dados relacional quanto a origem de dados do cubo para o DB2 Alphablox.

Instalando os Dados de Amostra para Sybase

Dados de amostra são fornecidos para origens de dados Sybase. É necessário instalar as tabelas de amostra para que seja possível utilizá-las para construir um cubo Alphablox requerido para o primeiro tutorial.

Para instalar os dados de amostra da QCC (Quality Chocolate Company) para origens de dados Sybase:

1. Copie todos os arquivos do diretório `sampledata/qcc/sybase` do CD de instalação do Alphablox para um diretório.
2. Execute o arquivo `qcc_syb.bat` com os parâmetros de sua conexão Sybase. Por exemplo, no prompt do DOS, digite `qcc_syb server1 blox dbo sa`. Esse comando executa o arquivo em lote e conecta-se ao Sybase no servidor `server1`,

no banco de dados blox como o proprietário de tabela dbo e nome de usuário sem senha. Os arquivos de registro a seguir são criados quando você executa o arquivo em lote:

- qcc.log para criação das tabelas da qcc
- qcc_fact.log para carregamento dos dados na tabela qcc_fact
- qcc_locations.log para carregamento dos dados na tabela qcc_locations
- qcc_products.log para carregamento dos dados na tabela qcc_products
- qcc_time.log para carregamento dos dados na tabela qcc_time

3. Consulte quaisquer mensagens nos arquivos de registro.

As tabelas são instaladas agora.

Nota: A tabela qcc será eliminada se ainda existir.

Continue com a seção “Definindo Origens de Dados” para definir tanto a origem de dados relacional quanto a origem de dados do cubo para o DB2 Alphablox.

Definindo Origens de Dados

Após a instalação das tabelas relacionais da QCC (Quality Chocolate Company) de seu banco de dados, defina duas origens de dados, uma para a origem de dados relacionais e outra para o cubo que irá criar.

Para definir origem de dados:

1. Crie a origem de dados qcc2003-rdb. Essa origem de dados será utilizada como origem de dados relacionais básica para seu cubo QCC.
 - a. Abra a página inicial do DB2 Alphablox em uma janela do navegador. A URL é `http://your server:port/AlphabloxAdmin/home`.
 - b. Selecione a guia **Administração > Origens de Dados**.
 - c. Clique em **Criar** para criar uma conexão de origem de dados para as tabelas da QCC.
 - d. No campo **Nome da Origem de Dados**, digite qcc2003-rdb.
 - e. Especifique os parâmetros de conexão apropriados para conectar-se às suas tabelas da QCC.
 - f. Clique em **Salvar**.
2. Crie a origem de dados qcc2003-acs. Essa origem de dados é para o cubo QCC.
 - a. Selecione **Administração > Origens de Dados** na página inicial do DB2 Alphablox.
 - b. Clique em **Criar** para criar uma conexão de origem de dados para o DB2 Alphablox Cube.
 - c. No campo **Nome da Origem de Dados**, digite qcc2003-acs e selecione **Alphablox Cube Server Adapter** a partir da lista **Adaptador**.
 - d. Clique em **Salvar**.

A última etapa para a construção de um cubo para os dados de amostra da QCC é armazenar o arquivo de definição de cubo fornecido no repositório do DB2 Alphablox.

- Se estiver utilizando um repositório com base em arquivos, siga as etapas da seção “Criando a Definição de Cubo de Amostra para um Repositório Baseado em Arquivos” na página 7.

- Se estiver utilizando um banco de dados como repositório, siga as etapas da seção “Criando a Definição de Cubo de Amostra para um Repositório Baseado em Banco de Dados”.

Criando a Definição de Cubo

Escolha as instruções para a criação da definição de cubo com base no tipo de repositório que está utilizando.

Criando a Definição de Cubo de Amostra para um Repositório Baseado em Arquivos

Após instalar os dados da QCC (Quality Chocolate Company) e definir as origens de dados de amostras, é necessário criar a definição de cubo.

Certifique-se de ter instalado os dados de amostra da QCC e que a origem de dados relacional e o cubo estejam definidos para o DB2 Alphablox.

Para criar uma definição de cubo da QCC:

1. No diretório *db2alphablox_dir/repository*, crie um diretório “cubes”.
2. No diretório *db2alphablox_dir/repository/cubes*, crie um diretório “qcc_2003”.
3. Copie o arquivo *qcc_2003.properties* do diretório *sampledata/qcc/acs* do CD para o diretório *alphablox_dir/repository/cubes/qcc_2003*.
4. Reinicie o servidor de aplicativos onde o DB2 Alphablox está sendo executado.

A definição do cubo está agora armazenada no repositório do DB2 Alphablox. O cubo está agora criado e definido para o DB2 Alphablox e está pronto para ser utilizado com os tutoriais e com o Blox Sampler.

Criando a Definição de Cubo de Amostra para um Repositório Baseado em Banco de Dados

Após instalar os dados da QCC (Quality Chocolate Company) e definir as origens de dados de amostras, é necessário criar a definição de cubo.

Certifique-se de ter instalado os dados de amostra da QCC e que a origem de dados relacional e o cubo estejam definidos para o DB2 Alphablox.

Para criar uma definição de cubo da QCC:

1. Crie um diretório temporário em seu sistema. Por exemplo, *c:\temp*.
2. Copie o arquivo *qcc_2003.properties* do diretório *sampledata/qcc/acs* do CD para o diretório temporário criado na Etapa 1.
3. Encerre o DB2 Alphablox, encerrando o servidor de aplicativos onde o DB2 Alphablox está sendo executado.
4. Execute o arquivo *ConvertRepository.bat* (ou *ConvertRepository.sh*) do diretório *db2alphablox_dir/bin/* para converter a definição de cubo armazenada no arquivo *qcc_2003.properties* de seu repositório de banco de dados.
5. Escolha a opção para convert from one repository to another (opção 3).
6. Escolha a opção para convert from file to database (opção 1).
7. Quando for solicitado que forneça a raiz do repositório, digite o caminho para o diretório temporário onde copiou o arquivo *qcc_2003.properties* na Etapa 2.
8. Siga os avisos e forneça os parâmetros de conexão apropriados.
9. Quando for solicitado para escolher a operação de conversão, digite U para atualizar o banco de dados.

10. Quando a conversão for concluída, reinicie o DB2 Alphablox iniciando o servidor de aplicativos onde o DB2 Alphablox está sendo executado.

O cubo está agora criado e definido para o DB2 Alphablox e está pronto para ser utilizado com os tutoriais e com o Blox Sampler.

Instalando Dados de Amostra Multidimensionais

Versões multidimensionais dos dados de amostra da QCC (Quality Chocolate Company) são fornecidos para uso com o Blox Sampler.

A instalação dos dados de amostra da QCC e sua preparação para uso com o Blox Sampler envolve duas etapas gerais:

1. Instale os dados de amostra. Selecione as instruções de instalação de seu banco de dados multidimensional.
 - DB2 OLAP Server V8.1 e Hyperion Essbase V6.5
 - DB2 OLAP Server V8.2 e Hyperion Essbase V7.1
 - Microsoft SQL Server 2000 Analysis Services
 - Microsoft SQL Server 2005 Analysis Services
2. Defina a origem de dados para o DB2 Alphablox.

Instalação de Dados de Amostra para DB2 OLAP Server V8.1 e Hyperion Essbase V6.5

Dados de amostra são fornecidos para origens de dados DB2 OLAP Server V8.1 e Hyperion Essbase V6.5. Esses dados de amostra da QCC (Quality Chocolate Company) podem ser utilizados com o Blox Sampler.

Para instalar os dados de amostra para origens de dados DB2 OLAP Server V8.1 e Hyperion Essbase V6.5:

1. Conecte-se ao DB2 OLAP Server ou Essbase Application Manager.
 - a. Inicie o DB2 OLAP Server ou Hyperion Essbase.
 - b. Abra o aplicativo Application Manager.
 - c. Selecione **Servidor** → **Conectar** e digite seu nome de login e senha.
 - d. Pressione **OK**.
2. Crie um aplicativo.
 - a. Selecione **Arquivo** → **Novo** → **Aplicativo**.
 - b. Digite Blox como o nome do aplicativo e Servidor como o local.
 - c. Clique em **OK**.
3. Crie um banco de dados.
 - a. Selecione **Arquivo** → **Novo** → **Banco de Dados**.
 - b. Digite qcc como o nome do banco de dados.
4. Abra e salve o esboço qcc.
 - a. Clique em **Abrir** para abrir o esboço.
 - b. Selecione **Arquivo** → **Abrir** a partir da janela Application Manager.
 - c. No campo **Local**, selecione **Cliente**.
 - d. Clique no botão **Sistema de Arquivos** e navegue para o arquivo de esboço qcc.otl em sampledata/qcc/essbase6/ no CD DB2 Alphablox.
 - e. Clique no botão **Abrir** para abrir o esboço.

- f. Feche a Janela de Esboço original. Esta janela não possui uma hierarquia desenvolvida.
 - g. Com a janela Novos Esboços aberta, selecione **Arquivo** → **Salvar Como** a partir da janela Application Manager.
 - h. No campo **Local**, selecione Servidor.
 - i. No campo **Aplicativo**, selecione blox.
 - j. No campo **Banco de Dados**, selecione qcc.
 - k. Clique em **OK**.
 - l. Feche a janela Esboço.
5. Carregue os dados da qcc.
 - a. A partir da janela do Application Manager, selecione **Banco de Dados** → **Limpar Dados** → **Todos**.
 - b. Selecione **Banco de Dados** → **Carregar Dados**
 - c. No campo **Arquivos de Dados**, selecione **Localizar**.
 - d. Na janela **Abrir Objetos de Arquivo de Dados do Servidor** aberta, selecione o botão de rádio **Cliente** na caixa **Local**.
 - e. Clique no botão **Sistema de Arquivos**.
 - f. Procure o arquivo qcc.txt em `sampledata/qcc/essbase6/` no CD do DB2 Alphablox e clique em **Abrir** para incluir o arquivo na lista de **Arquivos de Dados**.
 - g. Clique em **OK**. Os arquivos são carregados.
 6. Calcule o cubo.
 - a. Selecione **Banco de Dados** → **Calcular**.
 - b. Quando essa operação for concluída, saia do Application Manager.

O banco de dados de amostra está agora instalado.

Em seguida, defina a origem de dados do DB2 Alphablox seguindo as etapas em "Definindo a Origem de Dados" na página 13.

Instalação de Dados de Amostra para DB2 OLAP Server V8.2 e Hyperion Essbase V7.1

Dados de amostra são fornecidos para origens de dados DB2 OLAP Server V8.2 e Hyperion Essbase V7.1. Esses dados de amostra da QCC (Quality Chocolate Company) podem ser utilizados com o Blox Sampler.

Para instalar os dados de amostra para origens de dados DB2 OLAP Server V8.2 e Hyperion Essbase V7.1:

1. Conecte-se com o DB2 OLAP Server ou com o Essbase Administration Services
 - a. Inicie o DB2 OLAP Server ou Hyperion Essbase.
 - b. Abra o Console de Administração de seu DB2 OLAP Server ou Hyperion Essbase Administration Services.
 - c. Confirme que o DB2 OLAP Server ou Hyperion Essbase aparece no quadro de navegação abaixo de **Servidores Analíticos**.
2. Crie um aplicativo.
 - a. Na árvore de navegação abaixo do servidor, clique com o botão direito em **Aplicativos** e selecione **Criar Aplicativo** → **Utilizando Armazenamento em Bloco**.
 - b. Na janela Criar Aplicativo, digite Blox no campo **Nome do Aplicativo**.

- c. Clique em **OK**.
3. Crie um banco de dados.
 - a. Expanda o aplicativo Blox que acabou de criar.
 - b. Clique com o botão direito do mouse em **Bancos de Dados**, em seguida, selecione a opção **Criar Banco de Dados**.
 - c. Na janela Criar Banco de Dados, digite qcc e pressione **OK**.
 - d. Expanda **Bancos de Dados** e verifique se o novo banco de dados qcc existe.
4. Abra e salve o esboço qcc.
 - a. Na barra de menus do Console de Administração, selecione **Arquivo** → **Abrir**.
 - b. Na janela Abrir, clique na guia **Sistema de Arquivos** e procure o diretório `sampledata/qcc/essbase6` no CD do DB2 Alphablox.
 - c. Selecione o arquivo de esboço `qcc.otl` e clique no botão **OK**.
 - d. Confirme se o novo esboço é exibido no quadro direito.
 - e. Na barra de menus do Console de Administração, selecione **Arquivo** → **Salvar Como**.
 - f. Na janela Salvar Como, clique na guia **Servidor Analítico** e navegue para o diretório qcc.
 - g. Selecione o arquivo `qcc.otl` existente e clique em **OK**.
 - h. Na janela Confirmação, clique em **Sim** para substituir o arquivo existente.
 - i. Quando for perguntado se deseja abrir o esboço, clique em **Sim** para verificar se o novo esboço aparece.
5. Carregue os dados da qcc.
 - a. No Console de Administração, clique com o botão direito do mouse na lista de banco de dados da qcc e selecione a opção **Carregar Dados**.
 - b. Na janela Carregamento de Dados, clique no botão **Localizar Arquivo de Dados**. A janela Abrir aparece.
 - c. Clique na guia **Sistema de Arquivos** e procure o diretório `sampledata/qcc/essbase6` no CD do DB2 Alphablox.
 - d. Selecione o arquivo de dados `qcc.txt` e clique em **OK**.
 - e. Na janela Carregamento de Dados, confirme se o arquivo de dados está listado e clique em **OK**. A janela Carregamento de Dados Concluído aparece, mostrando que o arquivo de dados foi carregado sem erros.
 - f. Clique em **Fechar**.
6. Calcule o cubo.
 - a. No Console de Administração, clique com o botão direito do mouse na lista de banco de dados da qcc novamente, mas desta vez selecione a opção **Executar Cálculo**.
 - b. Na janela Executar Cálculo de Banco de Dados, verifique se o cálculo padrão está realçado e clique em **OK**. Quando o script de cálculo for concluído, você verá uma mensagem na janela Mensagens informando que o script de cálculo foi executado com êxito.
 - c. Feche o Console de Administração.

O banco de dados de amostra está agora instalado.

Em seguida, defina a origem de dados do DB2 Alphablox seguindo as etapas em “Definindo a Origem de Dados” na página 13.

Instalando Dados de Amostra do Microsoft SQL Server 2000 Analysis Services

Dados de amostra são fornecidos para origens de dados do Microsoft SQL Server 2000 Analysis Services. Esses dados de amostra da QCC (Quality Chocolate Company) podem ser utilizados com o Blox Sampler.

Para instalar os dados de amostra do Microsoft SQL Server 2000 Analysis Services:

1. Copie os arquivos Blox.CAB e qcc.mdb a partir de sampledata/qcc/msas no CD de instalação do DB2 Alphablox para um diretório em seu sistema (por exemplo, C:\Blox).
2. Clique no arquivo qcc.mdb com o botão direito do mouse e assegure-se de que a opção **Somente leitura** esteja desmarcada.
3. No Analysis Manager, restaure o cubo:
 - a. Clique com o botão direito do mouse no servidor de banco de dados e selecione **Restaurar Banco de Dados**.
 - b. Navegue para o arquivo Blox.CAB copiado na Etapa 1.
 - c. Selecione Blox.CAB e clique em **Abrir**. Se for perguntado se deseja sobrescrever o banco de dados existente, clique em **Sim**.
 - d. Na janela Restaurar Banco de Dados, clique em **Restaurar**.
 - e. Quando uma mensagem no final da janela indicar que o banco de dados foi restaurado com êxito, clique em **Fechar**.
4. Se seu arquivo qcc.mdb não estiver salvo em C:\Blox, que é o que o arquivo CAB espera, edite a origem de dados da seguinte forma:
 - a. Expanda o banco de dados do Blox.
 - b. Expanda **Origem de Dados**.
 - c. Clique com o botão direito do mouse na opção "C blox qcc mdb" e selecione **Editar**.
 - d. Selecione a guia **Conexão**.
 - e. Edite o caminho de seu arquivo qcc.mdb na caixa **Selecionar** ou digite um nome de banco de dados na caixa de edição.
 - f. Clique em **Testar Conexão** para certificar-se de que pode se conectar à origem de dados.
 - g. Clique em **OK**.
5. Verifique se a funcionalidade writeback está ativada para o cubo qccwb.
 - a. Expanda o banco de dados do Blox.
 - b. Expanda a pasta **Cubos**.
 - c. Clique no cubo qccwb com o botão direito do mouse e selecione **Ativar para Gravação** a partir do menu. Se a entrada **Opções de Writeback** estiver no menu exibido ao clicar com o botão direito do mouse, significa que a funcionalidade writeback está ativada.
 - d. Clique em **OK**.
 - e. Se a funcionalidade writeback não estiver ativada, clique novamente no cubo qccwb com o botão direito do mouse. A entrada **Opções de Writeback** deve estar disponível agora.

O banco de dados de amostra está agora instalado.

Em seguida, defina a origem de dados do DB2 Alphablox seguindo as etapas em "Definindo a Origem de Dados" na página 13.

Instalando Dados de Amostra do Microsoft SQL Server 2005 Analysis Services

Dados de amostra são fornecidos para origens de dados do Microsoft SQL Server 2005 Analysis Services. Esses dados de amostra da QCC (Quality Chocolate Company) podem ser utilizados com o Blox Sampler.

Para instalar os dados de amostra do Microsoft SQL Server 2005 Analysis Services:

1. Copie os arquivos qcc_ssas.abf e qcc_ssas.mdb a partir de sampledata/qcc/ssas2005 no CD de instalação do DB2 Alphablox para um diretório em seu sistema (por exemplo, C:\sampledata\qcc\ssas2005).
2. Clique no arquivo qcc_ssas.mdb com o botão direito do mouse e assegure-se de que a opção **Somente leitura** esteja desmarcada.
3. No SQL Server Management Studio, restaure o cubo:
 - a. Clique com o botão direito do mouse na pasta **Bancos de Dados** em seu servidor de banco de dados e selecione **Restaurar**.
 - b. Selecione a opção **Restaurar destino -> De arquivo de backup**.
 - c. No campo **Caminho Selecionado**, especifique o caminho para o arquivo qcc_ssas.abf.
 - d. Digite qcc_ssas como o nome do arquivo e clique em **OK**.
 - e. Digite password no campo **Criptografia de Senha** e clique em **OK**. Essa senha é requerida porque o arquivo de backup é criptografado.
 - f. Quando a restauração for concluída, clique com o botão direito do mouse na pasta **Bancos de Dados** e selecione **Atualizar** para ver o banco de dados MyBlox que acabou de ser criado.
 - g. Clique com o botão direito do mouse na pasta e renomeie o banco de dados de MyBlox para Blox.
4. Se o arquivo qcc_ssas.mdb não estiver salvo em C:\sampledata\qcc\msas\ssas2005, edite a origem de dados da seguinte forma:
 - a. Expanda o banco de dados do Blox.
 - b. Expanda **Origem de Dados**.
 - c. Clique com o botão direito do mouse no nome da visualização e selecione **Propriedades**.
 - d. Selecione o campo da cadeia de conexão na tabela para exibir o diálogo **Gerenciador de Conexão**.
 - e. No campo **Nome do Arquivo de Banco de Dados**, digite o caminho e o nome do arquivo e clique em **OK**.
5. Verifique se a funcionalidade writeback está ativada para o cubo qccwb.
 - a. Expanda o banco de dados do Blox.
 - b. Expanda a pasta **Cubos**.
 - c. Expanda o cubo qccwb.
 - d. Expanda os **Grupos de Medida**.
 - e. Expanda o cubo qccwb.
 - f. Clique com o botão direito do mouse na pasta **Writeback** e selecione a opção **Enable Writeback**.
6. Clique com o botão direito do mouse na pasta Blox, selecione **Processo** e clique em **OK**.

O banco de dados de amostra está agora instalado.

Em seguida, defina a origem de dados do DB2 Alphablox seguindo as etapas em “Definindo a Origem de Dados”.

Definindo a Origem de Dados

Quando os dados de amostra da QCC (Quality Chocolate Company) estiverem instalados em seu banco de dados, defina a origem de dados para o DB2 Alphablox.

Certifique-se de que os dados de amostra da QCC estejam instalados antes de continuar.

Para definir a origem de dados:

1. Vá para a página inicial do DB2 Alphablox. A URL é `http://your server:port/AlphabloxAdmin/home`.
2. Selecione **Administração > Origens de Dados**.
3. Clique em **Criar**.
4. Digite os seguintes valores para essa definição de origem de dados:

Tabela 1. Valores para a Definição de Origem de Dados da QCC

Campo	DB2 OLAP Server/Hyperion Essbase	Microsoft SQL Server Analysis Services
Nome da Origem de Dados	QCC-Essbase	QCC-MSAS
Adaptador	IBM DB2 OLAP Server ou Hyperion Essbase	OLE DB para OLAP
Nome do Servidor	<i>nome de seu servidor de dados IBM DB2 OLAP Server ou Hyperion Essbase</i>	<i>nome de seu servidor de dados Microsoft Analysis Services</i>
Aplicativo	Blox	
Banco de Dados	qcc	Blox
Nome do Usuário Padrão	<i>username</i>	<i>loginName</i>
Senha Padrão	<i>password</i>	<i>password</i>
Utilizar Nome do Usuário e Senha do DB2 Alphablox	Não	Não
Máximo de Linhas	1000	1000
Máximo de Colunas	1000	1000

5. Clique em **Salvar**.

A origem de dados está agora pronta para ser utilizada com o Blox Sampler.

Capítulo 2. Tutorial: Construindo Seu primeiro Aplicativo

Este tutorial explica conceitos básicos importantes sobre componentes Blox enquanto você cria um aplicativo analítico DB2 Alphablox simples.

Para construir aplicativos DB2 Alphablox, você irá utilizar a tecnologia JSP (JavaServer Pages), mas o conhecimento de JSP não é obrigatório para este tutorial. Este tutorial explica os aspectos da JSP que você precisa saber para começar a construir aplicativos analíticos DB2 Alphablox.

As etapas deste tutorial consideram que você está utilizando o DB2 Alphablox em um WebSphere Application Server. Se estiver utilizando o BEA WebLogic ou Apache Tomcat como servidor de aplicativos, pode ser necessário modificar algumas das etapas a seguir, conforme necessário para os diferentes servidores de aplicativos.

Pré-requisitos

O DB2 Alphablox deve estar instalado e acessível para você. Você precisa de um login Alphablox com direitos administrativos. Também é necessário que os bancos de dados de amostra do Alphablox estejam instalados, conforme descrito na tarefa Capítulo 1, "Instalando Bancos de Dados de Amostra no Alphablox", na página 1.

Definindo seu Aplicativo

Para criar um aplicativo utilizando uma abordagem de desenvolvimento J2EE, você precisa criar uma estrutura de diretórios com um diretório WEB-INF que contenha um arquivo do descritor de aplicativos (web.xml). O modo mais simples de criar esta estrutura no DB2 Alphablox é criar um novo aplicativo utilizando a página Aplicativo no DB2 Alphablox Admin Pages.

Para criar seu aplicativo e sua pasta:

1. Crie um novo aplicativo, denominado MyApp, no DB2 Alphablox.
 - a. Efetue login nas Páginas Administrativas do DB2 Alphablox como um usuário membro do grupo de administradores.
 - b. Clique na guia **Administração**.
 - c. Clique no link **Aplicativos**.
 - d. Clique no botão **Criar**. A página **Criar Aplicativo** é exibida.
 - e. Digite MyApp no campo **Nome**.
 - f. Digite My App (com um espaço) no campo **Nome da Exibição**, que define a etiqueta para o aplicativo que aparece na lista na página Aplicativos.
 - g. Clique no botão **Salvar** para definir o novo aplicativo e retornar para a página **Aplicativos**.

Nota: Se você estiver executando uma configuração Apache Tomcat com um servidor da Web externo (Apache ou IIS), deverá realizar as etapas necessárias em sua configuração para que o aplicativo se torne disponível. Para obter detalhes, consulte o *Guia do Administrador*.

2. Clique na guia superior **Aplicativos** no início da página à esquerda da página. Uma lista de aplicativos disponíveis é aberta.

3. Clique no nome de seu aplicativo criado (My App) na lista de aplicativos. Como você ainda não criou arquivos, o diretório está vazio. A pasta do novo aplicativo encontra-se no servidor de aplicativos (em um servidor WebSphere, a pasta está localizada no diretório WebSphere installedApps).

Você criou uma estrutura de aplicativo do DB2 Alphablox. Em seguida, pode incluir visualizações analíticas no aplicativo.

Acessando Dados

Você deve verificar se o banco de dados relacional e o Alphablox Cube Server estão em execução para que possa acessar a origem de dados de amostra do tutorial. O cubo do DB2 Alphablox também deve estar em execução.

Para verificar se suas origens de dados requeridas estão disponíveis e em operação:

1. Abra o navegador para o DB2 Alphablox Administration Pages.
2. Clique na guia **Administração**.
3. Clique no link **Origem de Dados**.
4. Na página Origens de Dados, localize a origem qcc2003-rdb no menu no lado esquerdo da página.
5. Clique em qcc2003-rdb na lista de origens de dados disponíveis e, em seguida, clique no botão **Testar Origem de Dados Seleccionada**. Se o banco de dados estiver em execução e a definição da origem de dados estiver configurada de maneira adequada, aparecerá uma mensagem de êxito. Se aparecer uma mensagem de erro, será necessário verificar a definição da origem de dados selecionando a origem de dados e clicando no botão **Editar** ou será necessário verificar se o banco de dados está configurado de maneira adequada e inclui um usuário com direitos de acesso ao banco de dados.
6. Selecione a origem de dados qcc2003-acs da lista de origens de dados e, em seguida, clique no botão **Testar Origem de Dados Seleccionada**. Se a origem de dados do Alphablox Cube Server estiver configurada de forma adequada, uma mensagem de êxito aparecerá. Se aparecer uma mensagem de erro, reveja a definição da origem de dados.

Você está preparado para aprender como construir visualizações analíticas simples do DB2 Alphablox.

Criando a Página Inicial do Aplicativo

Nesta lição, você irá criar uma página inicial simples utilizando habilidades básicas da Web. A página inicial irá conter links para acessar as visualizações analíticas criadas posteriormente neste tutorial.

Para fins deste tutorial, você só precisa criar uma estrutura de aplicativo da Web bastante simples. Na página inicial, é necessário criar dois links para acessar as duas visualizações analíticas criadas neste tutorial.

Para criar uma página inicial simples:

1. Utilizando um editor de texto ou sua ferramenta de implementação favorita, crie um arquivo denominado `index.html`, para sua página inicial. Para os propósitos deste tutorial, o arquivo pode ser muito simples.

2. Edite o arquivo para especificar um título do aplicativo (por exemplo, "My DB2 Alphablox Application") e dois links para as duas visualizações analíticas que serão criadas posteriormente neste tutorial.

Copie e cole, ou digite, o código a seguir no arquivo index.html.

```
<html>
<head>
<title>My DB2 Alphablox Application</title>
</head>
<body>
<h2>My DB2 Alphablox Application</h2>
<p>
<a href="PresentBloxView.jsp">Simple PresentBlox View</a>
</p>
<p>
<a href="ChartBloxView.jsp">Customized ChartBlox View</a>
</p>
</body>
</html>
```

3. Salve este arquivo no diretório MyApp criado ao definir o aplicativo.
4. Abra o navegador web na página inicial do DB2 Alphablox. Por padrão, o navegador exibe a página **Aplicativos**.
5. Se o DB2 Alphablox estiver instalado em um IBM WebSphere Application Server ou em um servidor BEA WebLogic, as listagens de diretórios deverão ser ativadas antes de você desempenhar essa etapa. Para instalações do DB2 Alphablox no Apache Tomcat, nenhuma etapa adicional deverá ser necessária. Se você vir um erro HTTP 403 (por padrão, as listagens de diretórios não são permitidas), poderá modificar o servidor para permitir listagens de diretórios ou apenas anexar o nome do arquivo myapp.html ao final da URL (por exemplo, http://localhost:9080/MyApp/myapp.html). A nova página deve ser exibida.
6. Clique no link **My App** para verificar se o arquivo é exibido no diretório. Clique no link **index.html** para visualizar sua home page.

Agora você tem uma página inicial com links para duas visualizações analíticas que serão criadas a seguir.

Configurando uma Página Inicial Padrão

Nas lições anteriores, você acessou a página inicial diretamente clicando no link do nome do arquivo no diretório de aplicativos MyApp ou especificando o arquivo da página inicial, index.html, na URL. Nesta lição, você especificará a página inicial padrão para o aplicativo.

Para configurar a página inicial padrão do aplicativo:

1. Abra seu navegador na página inicial do DB2 Alphablox.
2. Clique na guia **Administração**.
3. Clique no link **Aplicativos**.
4. Selecione **MyApp** na lista de aplicativos e clique no botão **Editar** abaixo da lista.
5. Na página Editar Aplicativo para MyApp, digite index.html no campo **URL Inicial**.
6. Clique no botão **Salvar** para salvar as alterações feitas.
7. Clique na guia principal de **Aplicativos**, na parte superior esquerda da guia da pasta, para retornar à página Aplicativos.

8. Clique no aplicativo **My App**. O aplicativo agora abre diretamente na página inicial definida.

Agora que a página inicial está definida, o servidor irá direcionar o aplicativo para abrir nesta página, mesmo que os usuários não especifiquem a página inicial no endereço da URL. Se um usuário digitar `http://yourServerName/MyApp/` na barra de endereços do navegador, o aplicativo abrirá automaticamente na página inicial definida, `index.html`.

Nas próximas lições, você começará a criar visualizações analíticas do DB2 Alphablox utilizando tags simples do Blox geradas utilizando o aplicativo DB2 Alphablox Query Builder.

Criando a Primeira Visualização Analítica

Nesta lição, você criará uma visualização analítica do DB2 Alphablox que exibe uma grade e um gráfico no componente PresentBlox. Na primeira visualização analítica que criar, você irá utilizar o aplicativo DB2 Alphablox Query Builder para gerar uma tag de customização JSP que define um componente PresentBlox.

Um componente PresentBlox pode incluir visualizações de grade e de gráfico, um painel de layout de dados, barras de ferramentas e outros componentes aninhados. As visualizações de grade e de gráfico representam visualizações alternativas sincronizadas dos mesmos dados.

Para criar sua primeira visualização analítica utilizando o DB2 Alphablox Query Builder:

1. Abra o navegador nas Páginas Administrativas do DB2 Alphablox. Por padrão, a guia Aplicativos fica visível.
 2. Clique no **DB2 Alphablox Query Builder** para abrir o aplicativo.
 3. Conecte-se ao banco de dados.
 - a. Clique no botão **Configurações de Conexão**. A janela Conexão do Banco de Dados abre.
 - b. Selecione a origem de dados `qcc2003-acs`. Pressione o botão **Conectar**. A janela será fechada e você verá Conectado na seção Status do Banco de Dados da página do Query Builder.
- Nota:** Não é necessário preencher os valores dos campos **Esquema**, **Catálogo**, **Nome do Usuário** e **Senha**, já que eles são recuperados automaticamente da definição da origem de dados para a origem de dados selecionada.
- c. Certifique-se de que a caixa de opções **Executar Consulta Padrão** esteja vazia.
4. Especifique a consulta.
 - a. Localize a seção Consulta. A seção Consulta está mais abaixo na página. Observe os seguintes recursos:
 - Para este tutorial, você está utilizando um cubo do DB2 Alphablox (`qcc2003-acs`). A seção Consulta contém uma lista de cubos disponíveis (atualmente em execução). O cubo `qcc2003-acs` deve ser listado.
 - Uma cadeia de consulta padrão associada a essa origem de dados aparece na janela de texto. Como a origem de dados é um cubo do DB2 Alphablox, o primeiro cubo, listado alfabeticamente, é incluído na instrução de consulta (por exemplo, `"select from [datasourceName]"`).

Restrição: Quando o recurso do conjunto de conexões no DB2 Alphablox Cube Server está ativado, não é possível utilizar o botão **Obter Consulta Padrão** para obter uma consulta padrão para um cubo do DB2 Alphablox.

- Se os dados aparecerem no componente PresentBlox na parte inferior da página, você pode trocar os eixos, executar drill up ou drill down, mover dimensões entre os eixos e assim por diante.
 - Para visualizar os resultados da consulta, clique no botão **Executar Consulta**. O conjunto de resultados aparece no componente PresentBlox na parte inferior da página.
- b. Digite qcc2003-ac3 entre os colchetes da consulta padrão (por exemplo, select from [qcc2003-ac3]) e, em seguida, pressione o botão **Executar Consulta**.
- A seção do PresentBlox deve exibir uma visualização padrão baseada na consulta.
5. Utilize a interface com o usuário do PresentBlox para criar uma nova visualização e uma nova consulta.
- a. Criar uma nova visualização.
- Por exemplo, você pode arrastar a dimensão Produtos para o eixo Linha no painel Layout de Dados à esquerda. Em seguida, arraste a dimensão Tempo (Calendário) para o eixo Coluna. Você pode então querer arrastar a dimensão Medidas para o eixo Página para que o menu Medidas apareça no painel Página. O painel Página exibe filtros de página acima do painel de exibição da grade e do gráfico. Selecione duas outras dimensões e inclua-as no eixo Página para que três filtros de página apareçam acima da exibição.
- b. Observe a instrução de consulta que aparece na seção Consulta. Se a consulta não estiver sendo gerada dinamicamente (quando a opção **Atualizar Automaticamente as Consultas** estiver vazia), você pode clicar no botão **Obter Consulta Atual** para obter a consulta atual. A caixa de texto exibe a instrução de consulta requerida para desenvolver a visualização de dados atual. Você pode copiar e colar as instruções de consulta do campo **Consulta** para serem utilizadas no desenvolvimento de aplicativos do DB2Alphablox.
6. Clique no botão **Gerar Tag Blox**. Uma janela é aberta mostrando a tag (e as tags aninhadas) necessária para reproduzir o layout do PresentBlox e o conjunto de resultados.
7. Copie a nova tag em um novo arquivo.
- a. Abra um novo arquivo no editor de texto.
- b. Inclua o seguinte texto ao arquivo.
- ```
<%@ taglib uri="bloxtld" prefix="blox" %>

//Copie e cole as informações sobre a tag da tag Blox gerada
//imediatamente abaixo desta linha. Remova todas as quebras de página da
//instrução de consulta. Você também pode remover esse comentário após
 incluir
//o código do DB2 Alphablox Query Builder.

<html>
<head>
<blox:header/>
</head>
<body>
<h2>Simple PresentBlox View</h2>
<p>
```

```

 <blox:display bloxRef="MyPresentBloxView"/>

 </p>
 </body>
</html>

```

- c. Substitua as linhas com comentário pelo código da etapa 6.
  - d. Salve o arquivo como PresentBloxView.jsp.
8. Copie o arquivo salvo (PresentBloxView.jsp) no diretório MyApp em seu servidor e teste a nova visualização analítica clicando no link Visualização Simples do PresentBlox da página inicial MyApp. Alternativamente, você pode digitar a URL diretamente em um endereço do navegador (por exemplo, <http://localhost:9080/MyApp/PresentBloxView.jsp>). A visualização que você vê deve ser semelhante ao componente PresentBlox que você manipulou no DB2 Alphablox Query Builder.

Um exemplo de código de uma versão completa da página Visualização Simples do PresentBlox (PresentBloxView.jsp) é mostrado abaixo:

```

<%@ taglib uri="bloxtld" prefix="blox" %>

<blox:present
 id="queryBuilder4_present"
 height="500"
 visible="false"
 width="100%">
 <blox:grid/>
 <blox:chart/>
 <blox:page/>
 <blox:data
 dataSourceName="qcc2003-acb"
 onErrorClearResultSet="true"
 query=" SELECT DISTINCT({[qcc_2003].[Time (Calendar)].[All Time
(Calendar)], [qcc_2003].[Time (Calendar)].[All Time (Calendar)].[2000],
[qcc_2003].[Time (Calendar)].[All Time (Calendar)].[2001],
[qcc_2003].[Time (Calendar)].[All Time (Calendar)].[2002],
[qcc_2003].[Time (Calendar)].[All Time (Calendar)].[2003]})
ON AXIS(0), DISTINCT({[qcc_2003].[Products].[All Products],
[qcc_2003].[Products].[All Products].[100 Truffles],
[qcc_2003].[Products].[All Products].[200 Chocolate Blocks],
[qcc_2003].[Products].[All Products].[300 Chocolate Nuts],
[qcc_2003].[Products].[All Products].[400 Specialties]})
ON AXIS(1) FROM [qcc_2003] WHERE ([qcc_2003].[Measures].[Sales],
[qcc_2003].[Time (Fiscal)].[All Time (Fiscal)],
[qcc_2003].[Date Opened].[All Date Opened],
[qcc_2003].[Has Nuts].[All Has Nuts],
[qcc_2003].[Chocolate Type].[All Chocolate Type],
[qcc_2003].[Ounces Per Package].[All Ounces Per Package],
[qcc_2003].[Pieces Per Package].[All Pieces Per Package],
[qcc_2003].[Date Introduced].[All Date Introduced],
[qcc_2003].[Seasonal].[All Seasonal],
[qcc_2003].[Scenario].[Actual],
[qcc_2003].[Locations].[All Locations]})"
 selectableSlicerDimensions="[qcc_2003].[Measures],
[qcc_2003].[Locations],[qcc_2003].[Chocolate Type]"
 useAliases="true"/>
 <blox:toolbar/>
 <blox:dataLayout/>
 <bloxui:calculationEditor />
</blox:present>
<html>
<head>
<blox:header/>
</head>

```

```

<body>
<h2>Simple PresentBlox View</h2>

<blox:display bloxRef="queryBuilder4_present"/>

</p>
</body>
</html>

```

No exemplo de código acima, a instrução de consulta tem quebras de linhas para melhorar a leitura. Para que esta página seja exibida corretamente, é necessário remover as quebras de linha dos valores de atributo (como os atributos query e selectableSlicerDimensions acima).

No código gerado pelo DB2 Alphablox Query Builder, muitas tags que estão incluídas não são necessárias. Vendo todas as tags aqui, no entanto, você deve ter uma percepção de como as tags Blox trabalham como pais e tags aninhadas. Além disso, você pode ver a utilização dos atributos de tags que são incluídos como resultado das manipulações feitas no componente PresentBlox do Query Builder.

O próximo tópico explicará muitos dos elementos importantes do código nesta página.

## Estrutura da Página de sua Primeira Visualização Analítica

Este tópico resume a estrutura da primeira visualização analítica (PresentBloxView.jsp) criada na lição anterior.

As tags Blox mais importantes utilizadas nesta página incluem: <blox:header/>, <blox:present> e <blox:display>. Essas três tags, e as tags aninhadas, especificam uma visualização analítica sem exigir nenhum código Java na página. A complexidade da lógica de apresentação está sendo gerenciada por essas tags Blox. Incluindo Blox aninhado e modificando os valores de atributos, você pode customizar visualizações com base nos requisitos de negócios.

Abaixo é apresentada uma pequena visão geral de alguns dos destaques da página criada na lição anterior. Para entender melhor o código, você precisará ler os tópicos de interesse no DB2 Alphablox Information Center.

O arquivo PresentBloxView.jsp começa com a seguinte linha:

```
<%@ taglib uri="bloxtld" prefix="blox" %>
```

Esta linha é uma diretiva de taglib JSP que informa ao servidor que você pretende utilizar uma biblioteca de tags Blox. O uri é um ponteiro para o arquivo do descritor da biblioteca de tags. O valor prefix, definido como blox, informa ao servidor para processar o conteúdo de todas as tags nesta página que iniciam com blox, utilizando a biblioteca de tags Blox, conforme definido no arquivo do descritor da biblioteca de tags.

Abaixo da diretiva taglib, o componente PresentBlox é especificado pelas seguintes tags e seus atributos de tag:

```

<blox:present
 id="queryBuilder4_present"
 height="500"
 visible="false"
 width="100%">
 <blox:grid/>
 <blox:chart/>
 <blox:page/>

```

```

<blox:data
 dataSourceName="qcc2003-acs"
 onErrorClearResultSet="true"
 query="SELECT DISTINCT({[qcc_2003].[Time (Calendar)].[All Time
(Calendar)], [qcc_2003].[Time (Calendar)].[All Time (Calendar)].[2000],
[qcc_2003].[Time (Calendar)].[All Time (Calendar)].[2001],
[qcc_2003].[Time (Calendar)].[All Time (Calendar)].[2002],
[qcc_2003].[Time (Calendar)].[All Time (Calendar)].[2003]})
ON AXIS(0), DISTINCT({[qcc_2003].[Products].[All Products],
[qcc_2003].[Products].[All Products].[100 Truffles],
[qcc_2003].[Products].[All Products].[200 Chocolate Blocks],
[qcc_2003].[Products].[All Products].[300 Chocolate Nuts],
[qcc_2003].[Products].[All Products].[400 Specialties]})
ON AXIS(1) FROM [qcc_2003] WHERE ([qcc_2003].[Measures].[Sales],
[qcc_2003].[Time (Fiscal)].[All Time (Fiscal)],
[qcc_2003].[Date Opened].[All Date Opened],
[qcc_2003].[Has Nuts].[All Has Nuts],
[qcc_2003].[Chocolate Type].[All Chocolate Type],
[qcc_2003].[Ounces Per Package].[All Ounces Per Package],
[qcc_2003].[Pieces Per Package].[All Pieces Per Package],
[qcc_2003].[Date Introduced].[All Date Introduced],
[qcc_2003].[Seasonal].[All Seasonal],
[qcc_2003].[Scenario].[Actual],
[qcc_2003].[Locations].[All Locations])"
 selectableSlicerDimensions="[qcc_2003].[Measures],
[qcc_2003].[Locations],[qcc_2003].[Chocolate Type]"
 useAliases="true"/>
<blox:toolbar/>
<blox:dataLayout/>
<bloxui:calculationEditor />
</blox:present>

```

A tag `<blox:present>` especifica que você quer que o `PresentBlox` apareça aqui com um valor de atributo `id` do `queryBuilder4_present`. O atributo `id` permite a você identificar este `Blox` particular para propósitos de script. Observe que a tag `<blox:present>` inclui um atributo `visible` configurado como `false`. Quando configurado como `false`, o `PresentBlox` não será renderizado até que a tag `<blox:display>` seja encontrada pelo compilador na página.

Existem muitos atributos disponíveis para cada `Blox` definido utilizando tags, mas eles devem ser incluídos apenas se você precisa especificar valores de atributos diferentes dos valores padrão. Neste exemplo, um valor `height` de 500 e um valor `width` de 100%. A altura e a largura podem ser definidas em pixels ou em porcentagens.

Um `DataBlox` aninhado é incluído, que define o atributo `dataSourceName` como `qcc2003-acs`. Se nenhuma origem de dados for especificada, você verá a mensagem "Nenhum dado disponível" quando a página for exibida. O atributo de consulta inclui a instrução de consulta MDX gerada pelo DB2 Query Builder. Ao compreender a linguagem de consulta MDX utilizada com o DB2 Alphablox Cube Server, você pode simplificar a consulta, tornando-a bem menor. A instrução de consulta aqui, no entanto, é completa e irá gerar a visualização criada no aplicativo Query Builder.

A seção `<head>` da página contém uma tag `Blox` especial para inclusão de código importante para a página, antes que ela seja renderizada:

```
<blox:header/>
```

Esta tag é utilizada pelo DB2 Alphablox para incluir automaticamente o HTML requerido, o JavaScript, e o código CSS na seção principal da página. Quando uma página é renderizada pelo servidor, esta tag inclui links CSS para temas HTML

definidos e uma tag meta para evitar armazenamento em cache. Lembre-se de digitar esta tag em cada página JSP que utiliza componentes Blox. Se esta tag não for incluída, a página não é renderizada corretamente.

A seção <body> da página contém a seguinte linha:

```
<blox:display bloxRef="MyPresentBloxView"/>
```

Essa tag Blox faz com que o componente PresentBlox especificado acima que seja renderizado aqui. Utilizar a tag <blox:display> permite que você faça alterações em um componente Blox ou em seus componentes aninhados em um local, onde também é possível ter outras tags Blox e scriptlets JSP controlando o comportamento dos componentes Blox. Você pode optar por ter um componente DataBlox que seja separado do componente PresentBlox e inclua scriptlets JSP que modificam os componentes do Blox e seu comportamento antes que o componente do Blox seja renderizado no navegador da Web.

Para obter visualizações e aplicativos mais complexos, os componentes adicionais do JavaBeans e código Java podem ser requeridos.

---

## Criando uma Visualização Analítica Customizada

Nesta lição, você irá criar uma visualização analítica customizada exibindo apenas um gráfico com os filtros de página.

Em muitos aplicativos, os dados são representados utilizando apenas uma grade ou um gráfico. O componente PresentBlox combina os componentes GridBlox e ChartBlox como componentes aninhados que são exibidos juntos no componente PresentBlox, mas você pode desativar ou ocultar os componentes aninhados que você não quer que sejam vistos. Para utilizar o componente PresentBlox para criar uma visualização de gráfico com filtros de página, modifique os atributos da tag de algumas das tags do Blox aninhadas.

Nesta tarefa, você reutilizará a tag do PresentBlox da primeira visualização analítica para criar uma segunda visualização analítica customizada.

Para criar uma visualização analítica que mostre um gráfico com filtros de página:

1. Abra o arquivo PresentBloxView.jsp criado anteriormente em seu editor e salve como ChartBloxView.jsp.
2. Nos componentes aninhados da tag PresentBlox listadas abaixo, inclua um atributo `visible` e configure os valores para `false`. Incluir o atributo fará com que os componentes aninhados especificados não sejam exibidos e deixará uma visualização de gráfico com uma barra de filtro de página.

```
<blox:grid visible="false"/>
<blox:chart/>
<blox:toolbar visible="false"/>
<blox:page/>
<blox:dataLayout visible="false"/>
```

3. Modifique a tag PageBlox (<blox:page>), incluindo o atributo `labelPlacement` e configurando o valor para `top`. Por padrão, o rótulo para os filtros de página aparecem à esquerda dos menus. A inclusão desse atributo e sua configuração para `top` substitui o comportamento padrão e colocará os rótulos acima dos filtros de página.

```
<blox:page labelPlacement="top" />
```

4. Modifique o nome do atributo `id` na tag `<blox:present>` para o seguinte valor: `chartview`. Cada componente do Blox exibido ou renderizado exige um ID exclusivo para evitar erros.
5. Salve suas alterações e inclua o arquivo em seu diretório `MyApp` do servidor.
6. Abra seu navegador e teste a nova página, clicando no link a partir de sua página inicial ou acessando a página diretamente através do endereço da URL (por exemplo, `http://localhost:9080/MyApp/ChartBloxView.jsp`)

Agora você criou duas visualizações analíticas utilizando o aplicativo DB2 Alphablox Query Builder, que o ajudou a começar. Além disso, você aprendeu que modificando ou incluindo atributos em tags existentes, você pode controlar a aparência e o comportamento do Blox e de seus componentes aninhados.

---

## Resumo

Se você concluiu todas as lições do tutorial, aprendeu como construir um aplicativo básico do DB2 Alphablox utilizando as Páginas Administrativas do DB2 Alphablox, o aplicativo DB2 Alphablox Query Builder e a biblioteca de tags do Blox. Várias propriedades e métodos estão disponíveis para definição e manipulação de componentes do Blox em páginas JSP. Desta forma qual é o próximo passo?

Se você deseja ver seus dados corporativos em uma visualização do DB2 Alphablox, você sabe agora como criar um aplicativo e incluir visualizações analíticas no mesmo. Se você deseja ver algo imediatamente, você deve fazer algumas modificações simples no aplicativo `MyApp`.

Para exibir dados de seus bancos de dados corporativos em vez de utilizar a origem de dados `qcc2003-acs`, você pode criar origens de dados que apontem para os bancos de dados corporativos, modificar os atributos `dataSourceName` para apontar para a origem de dados recém-definida e, em seguida, incluir um atributo `query` apropriado. Utilizando o DB2 Alphablox Query Builder, você pode criar rapidamente instruções de consulta utilizáveis e tags Blox.

---

## Capítulo 3. Tutorial: Construindo seu Primeiro Portlet com Componentes Blox

Neste tutorial, você aprenderá como incluir componentes Blox em seus portlets. Você:

1. Instalará um portlet de amostra pré-construído. Com esta etapa, você pode rapidamente ver como um componente Blox é incluído a uma página JSP do portlet e como ele é exibido em uma página do portal.
2. Escreverá sua própria página JSP com um GridBlox.

As tarefas neste tutorial não fornecem detalhes sobre o desenvolvimento geral de portlets. Este tutorial foca nas tarefas específicas do DB2 Alphablox e assume que exista alguma familiaridade com os conceitos gerais de ambientes de portais e desenvolvimento de portlets.

### Pré-requisitos

- O DB2 Alphablox deve estar instalado no servidor WebSphere Portal Versão 5.1. Consulte o *Guia de Instalação* para maiores detalhes sobre a instalação.
- O servidor WebSphere Portal deve estar iniciado.
- Você deve ter acesso administrativo para o servidor WebSphere Portal.
- Você deve ter familiaridade com as funções administrativas e interface com o usuário no WebSphere Portal.
- Você deve ter conhecimento básico de Java e JSP.
- Você deve ter um editor JSP instalado.

Embora você possa utilizar qualquer editor JSP ou um editor de texto para este tutorial, quando desenvolver seus próprios portlets, utilize uma ferramenta de desenvolvimento recomendada pelo WebSphere Portal, como Rational Application Developer.

Este tutorial utiliza uma origem de dados pré-definida instalada com o DB2 Alphablox. Você pode utilizar esta origem de dados para rapidamente desenvolver um aplicativo básico. Não se preocupe em configurar uma origem de dados personalizada para este tutorial.

---

## Instalando Portlets de Amostra

A melhor forma de aprender como incluir um Blox em seu portlet, é instalar os portlets de amostra fornecidos com o DB2 Alphablox e carregá-los em uma página do portal. Isso permite examinar a estrutura básica do código JSP e corresponder essa estrutura à saída do portal.

Instale os portlets de amostra fornecidos com o DB2 Alphablox:

1. Abra o navegador e efetue logon no portal como um usuário administrativo (a URL está no formulário do `http://<seu_servidor_de_portal>:<porta>/wps/portal`).
2. Clique no botão **Administração**.
3. Na seção Gerenciamento de Portlet, clique em **Módulos da Web**. A página Gerenciar Módulos da Web aparece à direita.

4. Clique em **Instalar**. Você recebe um aviso para instalar o Módulo da Web.
5. Clique no botão **Navegar** e navegue para o diretório `installableApps` no diretório de instalação do DB2 Alphablox.
6. Escolha o `AlphabloxSamplePortlets.war` e clique em **Avançar**. O aplicativo DB2 Alphablox Sample Portlets com um portlet chamado DB2 Alphablox JSP Page Sample Portlet é exibido na tabela de aplicativos Portlet.
7. Clique em **Concluir**.

O aplicativo DB2 Alphablox Sample Portlets e os portlets incluídos estão agora instalados. Selecione o diretório `installedApps` na instalação do WebSphere Portal. O nome do diretório recém-criado inicia com DB2 Alpha e termina com um ID do portlet gerado dinamicamente no padrão de `_PA_x_x_xx.ear`.

## Executando os Portlets de Amostra

Para executar os portlets de amostra na página do portal:

1. Vá para a página do portal.
2. Crie ou edite uma página existente. Você pode criar uma nova página para testar este portlet de amostra ou pode clicar em uma página existente para editar a página.
3. Clique em um dos botões **Incluir Portlets** na página de layout do portal.
4. Digite DB2 no campo Procurar e pressione **Procurar**. A caixa de opções **Portlet de Amostra da Página JSP do DB2 Alphablox** aparece.
5. Selecione a caixa de opções e clique em **OK**.
6. Clique em **Concluído**.

Quando a página do portal é atualizada, um `PresentBlox` irá aparecer. Esse `PresentBlox` possui:

- Uma barra de menu no início da página
- Dois painéis da barra de ferramentas debaixo da barra de menus
- Um painel de layout de dados na parte esquerda, que lhe permite mover as dimensões pelos diferentes eixos
- Uma grade que mostra os dados em um formato tabular
- Um gráfico de barras tridimensional à direita

Este arquivo `present.jsp` é a página padrão a ser carregada, como especificado no servlet `BloxJSPPagePortlet` incluído na amostra. O código fonte para esse servlet de amostra está disponível no diretório `WEB-INF/src/`.

Você instalou com sucesso os portlets de amostra e incluiu um portlet com `Blox` na página do portal. A próxima tarefa é examinar a estrutura de código nesta página JSP.

## Examinando a Estrutura de uma Página JSP do Portlet com um Componente Blox

Nesta tarefa, você revê a estrutura de código de uma página JSP que contém um `Blox`. Todas as páginas JSP devem ter os mesmos elementos chave.

Para abrir um arquivo JSP:

1. Navegue para o diretório `installedApps/` na instalação do WebSphere Portal, e localize a pasta de aplicativos criada mais recentemente que inicia com DB2 Alpha.



2. Navegue para o diretório PA\_x\_x\_xx.war/jsp/html/.
3. Abra o present.jsp no JSP ou no editor Java.
4. Examine o seguinte código e note os elementos chave:

```
<%@ page contentType="text/html"%>

<%@ taglib uri="bloxtld" prefix="blox"%>
<%@ taglib uri="/WEB-INF/tld/portlet.tld" prefix="portletAPI" %>

<portletAPI:init/>

<%
String bloxName = portletResponse.encodeNamespace("presentBlox");
%>

<head>
 <blox:header />
</head>

<blox:present id="presentBlox" bloxName="<%= bloxName %%" width="800">
 <blox:data dataSourceName="canned" />
</blox:present>
```

Este bloco de código contém seis elementos chaves:

1. A primeira linha informa ao navegador que a saída é HTML:

```
<%@ page contentType="text/html"%>
```

2. O próximo conjunto de código especifica as duas bibliotecas de tags JSP utilizadas nesta página:

```
<%@ taglib uri="bloxtld" prefix="blox"%>
<%@ taglib uri="/WEB-INF/tld/portlet.tld" prefix="portletAPI" %>
```

O uri é um ponteiro para o local do diretório onde o arquivo do descritor da biblioteca de tags está localizado. Os valores prefix, definidos como blox e portletAPI, informam ao servidor para:

- Procure por tags nesta página que comecem com blox e, em seguida, processe o conteúdo utilizando a Biblioteca de Tags do Blox como definido no arquivo do descritor da biblioteca de tags.
- Procure por tags nesta página que comecem com portletAPI e, em seguida, processe o conteúdo utilizando a Biblioteca de Tags do Portlet como definido no arquivo do descritor da biblioteca de tags.

3. Em seguida, uma tag de inicialização do portlet é incluída:

```
<API_do_potlet:init/>
```

Esta tag fornece acesso aos objetos PortletRequest, PortletResponse e PortletConfig. Com o PortletResponse, você pode chamar o método encodeNamespace() para assegurar que o nome do Blox não conflite com outros objetos em outros portlets executados na mesma página.

4. A próxima tag codifica o espaço de nomes para o Blox a incluir na página:

```
<%
String bloxName = portletResponse.encodeNamespace("presentBlox");
%>
```

Isto permite a você criar um Blox e designar esse nome exclusivo a ele.

5. O próximo bloco de código inclui uma tag de cabeçalho do Blox que é requerida para renderização do Blox e comunicação entre o servidor e o cliente:

```
<head>
 <blox:header />
</head>
```

Esta tag é utilizada pelo DB2 Alphablox para incluir automaticamente o HTML requerido, JavaScript, e código CSS na seção de cabeçalho da página. Quando uma página é renderizada pelo servidor, esta tag resulta na inclusão de links CSS aos temas HTML e uma tag meta para evitar armazenamento em cache. Esta tag deve ser incluída a cada página JSP que contém componentes Blox ou os componentes não serão renderizados corretamente.

6. Inclua um PresentBlox, utilizando tags fornecidas na Biblioteca de Tags Blox:

```
<blox:present id="presentBlox" bloxName="<%= bloxName %>" width="800">
 <blox:data dataSourceName="canned" />
</blox:present>
```

- Este código inclui um PresentBlox com um id do presentBlox e um bloxName do xx\_x\_x\_xxx\_presentBlox, que é o resultado da codificação do espaço de nomes.
- Esse Blox possui uma largura de 800 pixels e a altura de 400 pixels (o padrão).
- Um DataBlox aninhado é incluído, com o atributo tag dataSourceName configurado para canned.

O atributo da tag id do PresentBlox é requerido. Ele especifica o nome do script Java a ser utilizado na página JSP. O atributo bloxName especifica o nome do objeto no servidor. Um bloxName codificado assegura que a instância é exclusiva no servidor.

**Nota:** Se nenhuma origem de dados for especificada, você verá a mensagem **Nenhum dado disponível** na grade. A origem de dados canned é pré-definida durante a instalação e inclui uma pequena quantidade de dados de amostra. Ela não requer a instalação e configuração de um banco de dados externo real e pode ser utilizada para resolução de problemas e aprendizado. Como não é necessário especificar uma consulta, o atributo query não existe no código.

Esta página não tem uma tag outmost <html> pois esta página JSP é exibida em uma página do portal com outros portlets. Nenhuma tag <html> ou <body> adicional é necessária.

Agora que você examinou a estrutura do JSP e aprendeu o código essencial para incluir as páginas JSP do portlet, na próxima tarefa você criará uma nova página JSP com um Blox diferente e especificará alguns atributos Blox comuns para que a página se ajuste melhor na página do portal.

---

## Criando sua Própria Página JSP do Portlet com Componentes Blox

Nesta tarefa, você criará uma nova página JSP com um GridBlox e configurará algumas de suas propriedades. O objetivo é que você fique familiarizado com o constructo da tag geral do Blox durante a criação de um GridBlox que seja bem ajustado em uma página típica de portal.

Um GridBlox possui um tamanho padrão de 400x400 pixels. Ele vem também com uma barra de menus e uma barra de ferramentas. Você configurará algumas das propriedades comuns do GridBlox utilizadas, portanto, a grade utiliza apenas até

100 pixels de altura, com a barra de menus e a barra de ferramentas desativadas. Isso é feito, configurando as seguintes propriedades do GridBlox:

- `height` : configurado como 100 pixels
- `menubarVisible`: configurado como falso
- `toolbarVisible`: configurado como falso

Verifique se você atendeu aos requisitos especificados no Capítulo 3, "Tutorial: Construindo seu Primeiro Portlet com Componentes Blox", na página 25 e instalou o aplicativo de portlet de amostra, conforme descrito na seção no "Instalando Portlets de Amostra" na página 25.

Siga as etapas a seguir:

1. Na janela do navegador onde o PresentBlox é exibido, clique no botão Editar do portlet (o botão com um ícone de lápis). Uma lista drop-down de seleção é apresentada.

2. Na lista drop-down, selecione "Grid Blox" e clique em **OK**.

Conforme a página é atualizada, você deve ver um GridBlox de 400x400. Esse GridBlox, por padrão, possui a barra de menus e a barra de ferramentas ativas. Você alterará seu tamanho para 400x100, com a barra de menus e a barra de ferramentas desativadas.

3. Navegue para o diretório `PA_x_x_xx.war/jsp/html/` no aplicativo DB2 Alphablox Sample Portlets instalado anteriormente.

4. Abra `grid.jsp` no editor JSP. Esta página é quase idêntica ao `present.jsp`, exceto que:

- a tag `<blox:present>` agora indica que o `<blox:grid>` e o valor do `id` são diferentes:

```
<blox:grid id="gridBlox" bloxName="<%= bloxName %>" width="400">
 <blox:data dataSourceName="canned" />
</blox:grid>
```

- o valor de `bloxName` é diferente:

```
<%
 String bloxName = portletResponse.encodeNamespace("gridBlox");
%>
```

5. Configure a altura desse GridBlox como 100 pixels, incluindo o atributo `height` e configurando seu valor como 100:

```
<blox:grid id="gridBlox" bloxName="<%= bloxName %>" width="400" height="100">
 <blox:data dataSourceName="canned" />
</blox:grid>
```

6. Desative a barra de menus e a barra de ferramentas na parte superior, configurando os atributos `menubarVisible` e `toolbarVisible` como `false`:

```
<blox:grid id="gridBlox" bloxName="<%= bloxName %>" width="400" height="100"
 menubarVisible="false" toolbarVisible="false" >
 <blox:data dataSourceName="canned" />
</blox:grid>
```

Verifique se você digitou os nomes dos atributos corretamente, incluindo o uso de maiúsculas e minúsculas ("V" maiúsculo nos dois nomes de atributos). Verifique também se os atributos estão incluídos antes dos colchetes angulares de finalização (">").

7. Altere o espaço de nomes para `myFirstGrid`:

```
<%
 String bloxName = portletResponse.encodeNamespace("myFirstGrid");
%>
```

A alteração do espaço de nomes para esse GridBlox é para garantir que as alterações feitas sejam refletidas conforme você carrega esse JSP no portal.

Como você carregou essa página anteriormente, já existe uma instância desse GridBlox executando no servidor para essa sessão. A menos que você altere o espaço de nomes, as alterações feitas não serão refletidas mesmo que você atualize a página. A alteração do espaço de nomes é uma maneira rápida de testar suas alterações em um ambiente de desenvolvimento. De maneira alternativa, também é possível abrir uma nova janela do navegador para que um novo objeto seja criado no servidor para a nova sessão.

#### 8. Salve o arquivo.

Agora você está pronto para testar esse arquivo no WebSphere Portal.

Para testar as alterações feitas:

1. Volte para a página do portal.
2. Clique no botão Atualizar do Navegador para recarregar a página.

Você deve ver um GridBlox de 400x100 sem nenhuma barra de menus ou barra de ferramentas.

**Nota:** Essa lista drop-down de seleção do JSP não é uma funcionalidade interna do WebSphere Portal, mas criada por esse portlet de amostra. Selecione o arquivo `edit.jsp` e os arquivos de origem Java no diretório `WEB-INF/src/`.

---

## Criando um Projeto de Portlet para Utilizar Componentes Blox

Ao criar seu projeto de portlet na ferramenta de desenvolvimento, certifique-se de que o mapeamento de servlet do `AlphabloxServer` e as referências de biblioteca de tag necessárias para componentes do Blox sejam incluídas no arquivo `web.xml` de seu projeto, e que os arquivos descritores das Bibliotecas de Tags do DB2 `Alphablox` sejam copiados para seu projeto.

#### 1. Modifique o arquivo `web.xml` do projeto para incluir as seguintes linhas:

- Para obter a definição e o mapeamento do servlet:

```
<servlet>
 <servlet-name>AlphabloxServer</servlet-name>
 <servlet-class>com.alphablox.server.webapps.server.AlphabloxServer
 </servlet-class>
</servlet>

<servlet-mapping>
 <servlet-name>AlphabloxServer</servlet-name>
 <url-pattern>/abx/*</url-pattern>
</servlet-mapping>
```

- Para referências de biblioteca de tags (aplica-se apenas ao DB2 `Alphablox V8.4`; para usuários da `V8.4.1`, pule para a etapa 2):

**Importante:** Como a `V8.4.1` suporta o Servlet 2.4 ao invés do Servlet 2.2, seu servidor de aplicativos irá descobrir automaticamente os arquivos do descritor de biblioteca de tags. Se estiver utilizando a `V8.4.1`, não deve fazer referência explicitamente aos arquivos de descritor de referência da biblioteca de tags no arquivo `web.xml` de seu projeto.

```
<taglib>
 <taglib-uri>bloxtld</taglib-uri>
 <taglib-location>/WEB-INF/tlds/blox.tld</taglib-location>
</taglib>
<taglib>
 <taglib-uri>bloxformtld</taglib-uri>
 <taglib-location>/WEB-INF/tlds/bloxform.tld</taglib-location>
```

```

</taglib>
<taglib>
 <taglib-uri>bloxlogictld</taglib-uri>
 <taglib-location>/WEB-INF/tlds/bloxlogic.tld</taglib-location>
</taglib>
<taglib>
 <taglib-uri>bloxreporttld</taglib-uri>
 <taglib-location>/WEB-INF/tlds/bloxreport.tld</taglib-location>
</taglib>
<taglib>
 <taglib-uri>bloxportlettld</taglib-uri>
 <taglib-location>/WEB-INF/tlds/bloxportlet.tld</taglib-location>
</taglib>
<taglib>
 <taglib-uri>bloxuitld</taglib-uri>
 <taglib-location>/WEB-INF/tlds/bloxui.tld</taglib-location>
</taglib>

```

2. Copie os arquivos de descritor das Bibliotecas de Tags do DB2 Alphablox em seu projeto.
  - Na V8.4, copie os arquivos de descritor das Bibliotecas de Tags do DB2 Alphablox (arquivos TLD) a partir de *db2alphablox\_dir/bin* para o diretório WEB-INF/tlds de seu projeto.
  - Na V8.4.1, copie o arquivo *aastaglibs.jar* das Bibliotecas de Tags do DB2 Alphablox a partir de *db2alphablox\_dir/lib* para o diretório WEB-INF/lib de seu projeto.

## Configurando um Projeto de Portlet Utilizando o Rational Application Developer

O WebSphere Portal recomenda que você utilize o Rational Application Developer para desenvolvimento do portlet. Em particular, o Rational Application Developer Versão 6.0 é requerido para o WebSphere Portal Versão 5.1. O Rational Application Developer fornece um assistente que fornece instruções para a configuração e a criação de um projeto de portlet. Com base em sua seleção, ele configura a estrutura e o arquivo do descritor de implementação apropriados e cria automaticamente as classes Java necessárias para o controlador e páginas JSP para a visualização do portlet. Conforme você acompanha o assistente Novo Projeto de Portlet, verifique se o seguinte é configurado corretamente:

1. Escolha o "stub do WebSphere Portal v5.1" para o servidor de destino. Isso é especificado na primeira tela do assistente, clicando no botão **Mostrar Avançado >>** .
2. Inclua o mapeamento do servlet no DB2 Alphablox no arquivo *web.xml* criado no *WebContent/WEB-INF/*. Isso é feito digitando a seguinte definição de servlet e o código do mapeamento no *web.xml* do projeto:

```

<servlet>
 <servlet-name>AlphabloxServer</servlet-name>
 <servlet-class>com.alphablox.server.webapps.server.AlphabloxServer
 </servlet-class>
</servlet>

<servlet-mapping>
 <servlet-name>AlphabloxServer</servlet-name>
 <url-pattern>/abx/*</url-pattern>
</servlet-mapping>

```

3. Apenas para DB2 Alphablox V8.4 (para usuários da V8.4.1, pule para a etapa 4), inclua a Alphablox Tag Libraries no arquivo *web.xml* de seu projeto.

**Importante:** Como a V8.4.1 suporta o Servlet 2.4 ao invés do Servlet 2.2, seu servidor de aplicativos irá descobrir automaticamente os arquivos do descritor de biblioteca de tags. Se estiver utilizando a V8.4.1, não deve fazer referência explicitamente aos arquivos de descritor de referência da biblioteca de tags no arquivo de seu projeto.

- a. Com o `web.xml` aberto, clique na guia **Variáveis**.
- b. Role para a seção Referências a Bibliotecas de Tags na parte inferior.
- c. Clique em **Incluir**.
- d. Para **URL** e **Local**, digite os valores com base na seguinte tabela:

URL	Local
<code>bloxtld</code>	<code>/WEB-INF/tlds/blox.tld</code>
<code>bloxformtld</code>	<code>/WEB-INF/tlds/bloxform.tld</code>
<code>bloxlogictld</code>	<code>/WEB-INF/tlds/bloxlogic.tld</code>
<code>bloxreporttld</code>	<code>/WEB-INF/tlds/bloxreport.tld</code>
<code>bloxportlettld</code>	<code>/WEB-INF/tlds/bloxportlet.tld</code>
<code>bloxuitld</code>	<code>/WEB-INF/tlds/bloxui.tld</code>

Para cada par de URL e Local, é necessário clicar em **Concluir** para incluir a referência à biblioteca de tags e, em seguida, clicar em **Incluir** para incluir o próximo par.

4. Copie os arquivos de descritor das Bibliotecas de Tags do DB2 Alphablox em seu projeto.
  - Na V8.4, copie os arquivos de descritor das Bibliotecas de Tags do DB2 Alphablox (arquivos TLD) a partir de `db2alphablox_dir/bin` para o diretório `WEB-INF/tlds` de seu projeto.
  - Na V8.4.1, copie o arquivo `aastaglibs.jar` das Bibliotecas de Tags do DB2 Alphablox a partir de `db2alphablox_dir/lib` para o diretório `WEB-INF/lib` de seu projeto.

---

## Próximas Etapas

Você aprendeu como criar um portlet com componentes do Blox e a estrutura de código essencial. Para exibir dados de seus bancos de dados, é necessário:

1. Definir uma nova origem de dados para o DB2 Alphablox que aponte para a sua origem de dados.
2. Modificar o atributo `dataSourceName` do `DataBlox` para apontar para a origem de dados definida.
3. Incluir um atributo `query` com a cadeia de consulta no `DataBlox`.

Para obter detalhes sobre a criação de origens de dados, consulte o *Administrator's Guide*. Para aprender como criar consultas apropriadas, consulte *Retrieving Data* no *Developer's Guide* e a seção `DataBlox` do *Developer's Reference*.

Quando seus próprios dados aparecerem nas visualizações Blox, você também pode começar a explorar as muitas propriedades que podem ser configuradas, utilizando as tags do Blox.

---

## Dicas de Desenvolvimento de Portlets

Conforme você utiliza as informações fornecidas na documentação do DB2 Alphablox e continua o desenvolvimento do portlet, existem alguns tópicos específicos ao portlet e diretrizes gerais das quais você precisa estar ciente. O lista a seguir reúne essas diretrizes de desenvolvimento geral assim como ponteiros para seções específicas na documentação que são de seu interesse.

- Sempre codifique o nome do Blox que utiliza o espaço de nomes do portlet. Um espaço de nomes assegura um nome Blox exclusivo para a sessão J2EE atual.
- Sempre configure a largura e a altura de um Blox utilizando pixels. Fora do ambiente de portal, é possível configurar a largura e a altura para um valor de porcentagem como "50%" ou "100%". Os valores de porcentagem não funcionam no ambiente do portal, pois vários portlets coexistem na página.
- Não utilize URLs relativas para chamar os recursos em um portlet. As URLs devem ser codificadas utilizando o método `encodeURL()` da classe `PortletResponse`. Nos lugares aonde a documentação mostra o uso da URL relativa, você deve sempre codificar a URL.
- Para assegurar que o portlet do Blox utiliza um tema similar ao tema do portal, use o Portal Theme Utility. Este utilitário está disponível a partir de duas origens:
  - A página inicial do DB2 Alphablox.  
Na guia **Administração**, clique no link **Geral**. O Utility é listado sob a seção Portal. Consulte o *Administrator's Guide* e a ajuda on-line para obter detalhes.
  - O arquivo `AlphabloxAdminPortlets.war`.  
Essa é uma versão de portlet do Portal Theme Utility. Você pode instalar este portlet para executá-lo através do portal sem ter que efetuar login na página inicial do DB2 Alphablox.

Este utilitário combinará os estilos do ambiente do portal e os estilos no DB2 Alphablox, de modo que o Blox será exibido com cores e fontes semelhantes aos outros portlets na página.

- Selecione o tópico Blox Portlet Tag Library no *Developer's Guide* para obter os problemas que deseja levar em consideração durante a fase de planejamento.
- O DB2 Alphablox possui um Modelo eficiente de UI do Blox. O Modelo UI inclui um objeto `ClientLink` que lhe permite carregar uma URL especificada quando um componente Blox na página é clicado. Para o desenvolvimento do portlet, utilize a Biblioteca de Tags do Portlet Blox para criar o `ClientLink`. As tags manipularão a formação e a manipulação da URL de maneira dinâmica para que os links não se tornem stale depois que a página for atualizada. Consulte o tópico Planning for portlet development no *Developer's Guide* para obter detalhes.
- Para o portlet e ações de URIs, utilize a Biblioteca de Tags do Portlet Blox para criar o link do portlet ou da ação. Em seguida, é possível utilizar a API do Portlet do portal para processar a ação ou o link do portlet. Consulte o tópico Blox Portlet Tag Library no *Developer's Guide* para obter detalhes.





---

## Capítulo 4. Tutorial: Construindo Aplicativos Utilizando Ferramentas Rational Developer

Você pode utilizar as ferramentas Rational Developer com o DB2 Alphablox Toolkit para desenvolver mais rapidamente aplicativos utilizando componentes do DB2 Alphablox e as APIs Java do DB2 Alphablox.

Ferramentas Rational Developer (Rational Application Developer ou Rational Web Developer) podem ser utilizadas com o DB2 Alphablox Toolkit, um conjunto de plug-ins baseados no Eclipse, para desenvolver e testar aplicativos construídos utilizando a tecnologia do DB2 Alphablox. Este tutorial o orientará em como configurar o ambiente de desenvolvimento integrado do Rational para permitir o uso de conclusão de métodos e tags, assim como aprimoramentos customizados incluídos com o DB2 Alphablox Toolkit.

Para iniciar a configuração da ferramenta Rational Developer com o DB2 Alphablox Toolkit para desenvolver e testar os aplicativos do DB2 Alphablox.

---

### Preparando seu Ambiente de Desenvolvimento

Antes que você possa instalar e iniciar a utilização do DB2 Alphablox Toolkit, prepare seu ambiente de desenvolvimento certificando-se de que você tenha todos os softwares requeridos.

Para preparar o ambiente de desenvolvimento:

1. Instale uma ferramenta Rational Developer (Rational Application Developer ou Rational Web Developer) em sua estação de trabalho.
2. Instale quaisquer atualizações necessárias para assegurar que você tenha ao menos o Rational Application Developer ou o Rational Web Developer versão 6.0.0.1.
3. Se você planeja utilizar ambientes de teste integrados do WebSphere 5.1, certifique-se de que você atualizou a versão do Rational Developer para incluir a instalação deste opcional.

**Nota:** As instalações padrões das ferramentas Rational Developer não incluem os ambientes de teste integrados do WebSphere 5.1.

Você agora está pronto para executar as tarefas neste tutorial.

---

### Instalando o DB2 Alphablox Toolkit

A utilização do DB2 Alphablox Toolkit com a ferramenta Rational Developer facilitará o início do desenvolvimento de aplicativos baseado na web com o conteúdo do DB2 Alphablox.

O DB2 Alphablox Toolkit requer a utilização do Rational Application Developer ou do Rational Web Developer versão 6.0.0.1.

A instalação do DB2 Alphablox Toolkit em uma ferramenta Rational Developer poupa os desenvolvedores de muitas etapas manuais para permitir a utilização da conclusão do conteúdo para métodos Java e tags de customização JSP. Além disso,

assistentes customizados e folhas de dicas podem ser utilizados para orientá-lo na criação de aplicativos com o conteúdo do DB2 Alphablox e na definição de instâncias do servidor WebSphere.

Para instalar o DB2 Alphablox Toolkit no Rational Application Developer ou no Rational Web Developer:

1. Insira o Disco de Instalação do DB2 Alphablox na unidade de CD de sua estação de trabalho.
2. No diretório plugin, localize o subdiretório UpdateSite.
3. Copie o diretório UpdateSite em um local conveniente no disco rígido. Por exemplo, copie o diretório no C:\DB2Alphablox\UpdateSite.
4. Inicie a ferramenta Rational Developer.
5. Na barra de menus, selecione **Ajuda > Atualizações de Software > Localizar e Instalar**.
6. Na janela de Instalação que se abre, selecione a opção **Procurar por novos recursos a serem instalados** e, em seguida, clique em **Avançar**.
7. Na janela Sites de Atualização a Visitar, clique no botão **Novo Site Local** e navegue para o local do diretório UpdateSite.
8. Clique em **Avançar**, selecione o recurso **DB2 Alphablox Toolkit** e clique em **Avançar** novamente.
9. Na janela seguinte, aceite o contrato de licença e selecione o local aonde o recurso DB2 Alphablox Toolkit será instalado.
10. Depois da instalação ser concluída, reinicie a ferramenta Rational Developer

Depois de reiniciar a ferramenta Rational Developer, os recursos DB2 Alphablox Toolkit estarão disponíveis.

---

## Instalando o DB2 Alphablox nos Ambientes de Teste Integrados do WebSphere

Nesta tarefa, você instala o DB2 Alphablox nos ambientes de teste integrado do WebSphere em uma ferramenta Rational Developer.

**Pré-requisitos:** o ambiente de teste integrado do WebSphere suportado deve ser instalado na ferramenta Rational Developer. Você precisa acessar o Disco de Instalação do DB2 Alphablox. Revise e desempenhe quaisquer etapas requeridas de pré-instalação, descritas na seção Tarefas de Pré-instalação do *DB2 Alphablox Installation Guide*.

O DB2 Alphablox pode ser instalado em servidores de aplicativos WebSphere independentes ou em ambientes de teste integrados do WebSphere disponíveis na ferramenta Rational Developer. O resumo das etapas descritas abaixo explicam as diferenças na instalação ao instalar o DB2 Alphablox em tempos de execução do WebSphere na ferramenta Rational Developer.

Para instalar o DB2 Alphablox em ambientes de teste integrados do WebSphere

Instale o DB2 Alphablox seguindo as etapas na seção de Instalação do *Guia de Instalação do DB2 Alphablox*, com as seguintes exceções:

1. Na janela Configurar WebSphere do instalador, no campo **Diretório Raiz do WebSphere**, especifique o local do tempo de execução do WebSphere que você deseja utilizar como servidor de ambiente de teste integrado.

Por exemplo, para instalar o DB2 Alphablox no ambiente de teste integrado do WebSphere 6 na instalação padrão, selecione o caminho para o diretório base\_v6. Para uma instalação típica do Rational Application Developer, o caminho deve ser semelhante ao seguinte:

C:\Arquivos de programas\IBM\Rational\SDP\6.0\runtimes\base\_v6

2. Na janela Configurações do WebSphere, especifique um nome e senha para o administrador do WebSphere. Esses valores de entrada não são utilizados no ambiente de teste integrado do WebSphere, mas são requeridos pelo instalador do DB2 Alphablox.

**Importante:** Não desempenhe as etapas de pós-instalação descritas no *DB2 Alphablox Installation Guide*.

Depois de instalar o DB2 Alphablox no tempo de execução do WebSphere, você pode criar instâncias do servidor WebSphere para testar aplicativos DB2 Alphablox na ferramenta Rational Developer.

---

## Criando uma Instância do Servidor WebSphere

Para executar aplicativos ou arquivos JSP que contenham conteúdo do DB2 Alphablox, você precisa criar uma instância do servidor WebSphere que possa acessar serviços requeridos do DB2 Alphablox e classes Java.

**Pré-requisitos:** Configure a ferramenta Rational Developer. Instale o DB2 Alphablox Toolkit. Instale o DB2 Alphablox nos ambientes de teste integrados do WebSphere.

Para criar uma instância do servidor WebSphere para aplicativos de teste e arquivos JSP contendo conteúdo do DB2 Alphablox:

1. Abra a ferramenta Rational Developer.
2. Clique na guia **Servidores** (se a guia não estiver visível, selecione **Janela > Mostrar Visualização > Servidores**).
3. Clique com o botão direito do mouse dentro da janela de visualização e selecione **Novo > Servidor**. A janela Novo Servidor é aberta.
4. Digite localhost como o nome do host e selecione o tipo do servidor. Clique em **Avançar**.
5. Digite o número da porta do servidor e clique em **Avançar**.
6. Inclua os projetos disponíveis que deseja executar nesta instância do servidor na lista de projetos configurados. Clique em **Concluir**. A nova instância do servidor é aberta na visualização **Servidores**.

Se você criou uma instância do servidor WebSphere 5.1, você precisa executar a próxima etapa, "Criando Variáveis de Substituição do Servidor WebSphere 5.1." Para instâncias do servidor WebSphere 6, prossiga para "Incluindo Guests no Grupo de Administradores do DB2 Alphablox."

---

## Criando Variáveis de Substituição do Servidor WebSphere 5.1

Nesta tarefa, você modificará variáveis de substituição do servidor para instâncias do servidor WebSphere 5.1 para executar o DB2 Alphablox apropriadamente nos ambientes de teste integrado do WebSphere.

1. Abra a visualização **Servidores** e clique duas vezes na instância do servidor do WebSphere 5.1 que deseja modificar.

2. Clique na guia **Configuração** da instância do servidor.
3. Selecione **Ativar Console de Administração** e desmarque **Ativar Cliente de Teste Universal**.
4. Clique na guia **Variáveis** da instância do servidor. A janela **Variáveis de Substituição** é aberta.
5. Inclua duas novas variáveis utilizando o botão **Incluir** (localizado próximo à lista **Configurações do Nó**).
  - a. Inclua uma variável denominada `WS_EAR_AlphabloxPlatform` e configure o valor como `$(APP_INSTALL_ROOT)/localhost/AlphabloxPlatform.ear`.
  - b. Inclua uma segunda variável denominada `WS_EAR_AlphabloxStudio` e configure o valor como `$(APP_INSTALL_ROOT)/localhost/ApplicationStudio.ear`.
6. Salve as alterações.

Para instâncias do servidor do WebSphere 5.1, é necessário seguir as etapas em "Configurando as Instâncias do Servidor do WebSphere 5.1" para concluir a modificação da instância do servidor.

---

## Configurando as Instâncias do Servidor WebSphere 5.1

Para instâncias do servidor WebSphere 5.1, você precisa configurar as instâncias do servidor para executar aplicativos e arquivos do DB2 Alphablox.

**Pré-requisitos:** Crie as variáveis de substituição do servidor WebSphere 5.1 requeridas.

Para configurar as instâncias do servidor WebSphere 5.1:

1. Abra a visualização **Servidores** na ferramenta Rational Developer e inicie a instância do servidor que deseja configurar.
2. Clique com o botão direito na instância do servidor e selecione **Executar Console Administrativo**.
3. Não digite um valor no campo **ID**. Clique em **OK**.
4. Clique no botão **OK** abaixo da tabela de configuração.
5. Abra a visualização de navegação **Aplicativos Corporativos** e clique no nome do aplicativo **ApplicationStudio**.
6. Configure **Binários do Aplicativo** como `$(WS_EAR_ApplicationStudio)`.
7. Selecione **Utilizar Metadados de Binários**.
8. Clique no botão **OK** abaixo da tabela de configuração.
9. Clique no botão **Salvar** no WebSphere Administrative Console.
10. Na janela Salvar para Configuração Principal, clique em **Salvar**.
11. Inicie os aplicativos `AlphabloxPlatform` e `ApplicationStudio` no WebSphere Administrative Console em **Aplicativos > Aplicativos Corporativos**.

A instância do servidor está configurada. A seguir você precisa executar a etapa "Incluir Guest no Grupo de Administradores do DB2 Alphablox".

---

## Incluindo o Usuário Guest no Grupo de Administradores do DB2 Alphablox

Incluir o usuário guest ao grupo de Administradores do DB2 Alphablox permite que você acesse as Páginas de Administração do DB2 Alphablox ao utilizar a ferramenta Rational Developer.

**Pré-requisitos:** Crie a instância do servidor WebSphere. Configure a instância do servidor WebSphere.

Para incluir o usuário guest ao grupo de Administradores do DB2 Alphablox.:

**Importante:** Não conceda direitos de administrador ao usuário guest em seus servidores WebSphere de produção.

1. Efetue login no console do DB2 Alphablox utilizando o seguinte comando telnet: `telnet localhost portNumber` . em que *portNumber* é a porta especificada durante a instalação do DB2 Alphablox.
2. No prompt do console do telnet, digite o seguinte comando de console do DB2 Alphablox: `set Administrators guest` e pressione Enter.
3. No console do telnet, digite `Salvar` e pressione Enter.
4. Feche a sessão de telnet.

A instância do servidor WebSphere está agora pronta para utilização.

---

## Criando Aplicativos do DB2 Alphablox

Quando você cria novos aplicativos no Rational, é necessário incluir conteúdo requerido do DB2 Alphablox para que os aplicativos sejam executados de maneira apropriada.

**Pré-requisitos:** Instale o DB2 Alphablox Toolkit na ferramenta Rational Developer.

1. Na barra de menus da ferramenta Rational Developer, selecione **Arquivo > Novo Projeto**. O assistente **Novo Projeto** é aberto.
2. Expanda a opção **Web**, e selecione **Projeto da Web Dinâmico** e, em seguida, clique em **Avançar**.
3. Digite um nome para o projeto e clique no botão **Mostrar Avançado**. Opções adicionais aparecem.
4. Selecione a versão do servlet apropriado e o servidor de destino.
5. Clique em **Avançar**. A janela **Opções de Recursos** é aberta.
6. Selecione a opção **Conteúdo do DB2 Alphablox** e clique em **Concluir**.

Seu novo projeto de aplicativo da Web agora é ativado pelo DB2 Alphablox. As bibliotecas de tags do DB2 Alphablox e da API Java Blox estão disponíveis e o arquivo descritor do aplicativo (`web.xml`) foi modificado para incluir informações requeridas do DB2 Alphablox. Agora você pode começar a incluir arquivos JSP contendo o conteúdo do DB2 Alphablox para o projeto.

---

## Criando Arquivos JSP com o Conteúdo do DB2 Alphablox

Nesta tarefa, você cria novos arquivos JSP com acesso para as bibliotecas de tags do Blox utilizando a ferramenta Rational Developer com o DB2 Alphablox Toolkit.

**Pré-requisitos:** Instale o DB2 Alphablox Toolkit na ferramenta Rational Developer.

1. Na visualização **Explorador de Projetos** do Rational Developer, selecione **Arquivo > Novo > Arquivo JSP**.
2. Digite o nome do arquivo no campo **Nome do Arquivo**.
3. Clique em **Configurar Opções Avançadas** e pressione **Avançar**.
4. Inclua as bibliotecas de tags do DB2 Alphablox que você utilizará no arquivo JSP.
  - a. Clique no botão **Incluir** para abrir a janela Incluir Bibliotecas de Tags.
  - b. Selecione as bibliotecas de tags do DB2 Alphablox que planeja utilizar no arquivo JSP.
  - c. Clique em **Avançar**.
5. Selecione **ISO 10646/Unicode(UTF-8)** da lista **Codificação**. A codificação UTF-8 é requerida para que os aplicativos DB2 Alphablox sejam executados corretamente.
6. Clique em **Concluir**. O novo arquivo JSP aparece na listagem do projeto.
7. Dê um clique duplo no nome do arquivo para abrir o arquivo na janela do editor JSP.
8. Posicione o cursor na tag HTML <head>, mas depois das diretivas de taglib JSP do DB2 Alphablox incluídas anteriormente.
9. Digite a guia do cabeçalho Blox digitando o seguinte na linha nova:  
<blox:header/> A tag do cabeçalho Blox é necessária para incluir o DB2 Alphablox JavaScript e os arquivos CSS quando você executa o arquivo JSP.

O novo arquivo JSP está agora ativado para acessar as bibliotecas de tags do DB2 Alphablox que você selecionou. Com o recurso Assistente de Conteúdo da ferramenta Rational Developer, você pode inserir tags Blox e atributos de tags para as bibliotecas de tags selecionadas.

---

## Acessando Origens de Dados do DB2 OLAP Server e do Essbase

Para acessar as origens de dados do DB2 OLAP Server ou do Essbase na ferramenta Rational Developer ao utilizar instâncias do servidor WebSphere 5.1, você precisa criar um arquivo em batch de inicialização que carregue as bibliotecas do cliente requeridas.

**Pré-requisitos:** Instale o DB2 Alphablox Toolkit. Instale o DB2 Alphablox nos ambientes de teste integrados do WebSphere 5.1. Configure as instâncias do servidor WebSphere 5.1. Instale o IBM DB2 OLAP Server ou Hyperion Essbase na máquina de desenvolvimento.

Devido às limitações nos ambientes de teste integrados do WebSphere 5.1 na ferramenta Rational Developer, você precisa criar um arquivo em batch de inicialização para a ferramenta Rational Developer para acessar origens de dados do DB2 OLAP Server ou Hyperion Essbase. Ao iniciar a ferramenta Rational Developer utilizando o arquivo em batch, as bibliotecas do cliente do Essbase requeridas são incluídas no caminho da biblioteca Java.

Para criar um arquivo em batch para ativar o acesso ao DB2 OLAP Server ou Essbase com instâncias do servidor WebSphere 5.1:

1. Utilizando um editor de texto, crie um novo documento de texto.
2. Inclua uma linha de código para chamar o arquivo aassetup.bat do DB2 Alphablox, localizado no diretório de instalação do DB2 Alphablox. Por exemplo, o seguinte código executará o arquivo aassetup.bat localizado na instalação do DB2 Alphablox especificada:

```
call C:\alphablox\analytics\bin\aasetup.bat
```

3. Inclua uma segunda linha para chamar o arquivo `rationalSDP.exe` do RAD, que iniciará o RAD. Por exemplo, o seguinte código executará a ferramenta Rational Developer:

```
call C:\Arquivos de programas\IBM\Rational\SDP\6.0\rationalSDP.exe
```

4. Salve este arquivo como `startRAD.bat` no desktop da estação de trabalho (ou em outro local conveniente).

Ao dar um clique duplo no arquivo `startRAD.bat`, o `aasetup.bat` do DB2 Alphablox executa e configura os caminhos e variáveis de ambiente requeridos, em seguida, o Rational Application Developer é iniciado. A versão das bibliotecas do cliente Essbase utilizadas com o DB2 Alphablox precisam corresponder com a versão do DB2 OLAP Server (ou Hyperion Essbase) que você está utilizando como uma origem de dados. Para modificar as bibliotecas do cliente Essbase utilizadas com o DB2 Alphablox, execute o DB2 OLAP Server / Essbase Client Library Utility (`ChangeEssbase.bat`), que pode ser localizado no diretório `db2_alphablox\analytics\bin`, em que `db2_alphablox` é o diretório raiz da instalação do DB2 Alphablox. Este utilitário modifica um dos arquivos em batch executados quando você inicia o RAD utilizando o arquivo em batch `startRAD` que você criou.

Segue abaixo um exemplo completo do arquivo `startRAD.bat` descrito acima:

```
call C:\alphablox\analytics\bin\aasetup.bat
call C:\Arquivos de programas\IBM\Rational\SDP\6.0\rationalSDP.exe
```

Quando você pretende acessar as origens de dados do DB2 OLAP Server ou Essbase, o arquivo `startRAD.bat` deve ser utilizado para iniciar a ferramenta Rational Developer.





---

## Capítulo 5. Tutorial: Construindo um Cubo do DB2 Alphablox, Utilizando o DB2 Cube Views

O tutorial o orienta para a criação de um cubo DB2 Alphablox construído utilizando um banco de dados de amostra do DB2 Cube Views.

As tarefas no tutorial não fornecem detalhes sobre a construção de cubos DB2 Alphablox customizados. O objetivo é mostrar a você como criar rapidamente um cubo DB2 Alphablox que pode ser utilizado na exploração de recursos do DB2 Alphablox Cube Server. A origem de dados resultante pode também ser utilizada para testar e construir aplicativos DB2 Alphablox.

### Pré-requisitos:

- Instalar o DB2 Alphablox. Consulte o *Guia de Instalação* para maiores detalhes sobre a instalação.
- Ter direitos de acesso para uma implementação do DB2 Cubes Views suportada que tenha o banco de dados de amostra do DB2 Cube Views CVSAMPLE instalado. Consulte o *Guia de Instalação* para maiores detalhes sobre as versões suportadas do DB2 Cube Views.

No tutorial, você aprenderá como construir um cubo DB2 Alphablox baseado no banco de dados de amostra do DB2 Cube Views CVSAMPLE. Durante o tutorial, você aprenderá as seguintes tarefas:

---

### Definindo uma Origem de Dados Relacionais do DB2

Nesta tarefa, você definirá uma definição da origem de dados do DB2 Alphablox para um banco de dados DB2.

Pré-requisitos: Os drivers JDBC requeridos do DB2 devem estar acessíveis para o DB2 Alphablox.

Um cubo do DB2 Alphablox requer que a origem de dados relacionais básica seja predefinida como uma origem de dados do DB2 Alphablox. Os cubos do DB2 Alphablox são gerados utilizando os metadados e dados disponíveis nos bancos de dados relacionais.

Para definir uma origem de dados do DB2 Alphablox para um banco de dados DB2:

1. Efetue logon nas Páginas de Administração do DB2 Alphablox como o usuário admin (ou como um usuário com direitos de administrador).
2. Clique na guia **Administração**.
3. Clique no link **Origem de Dados**.
4. Clique no botão **Criar**.
5. No menu **Adaptador**, selecione o driver JDBC apropriado do IBM DB2 para seu servidor de banco de dados.  
Selecione **Driver JDBC Tipo 4 do IBM DB2** ou **Driver IBM DB2 UDB no iSeries**.
6. No campo **Nome da Origem de Dados**, digite CVSAMPLE como o nome a ser utilizado para a origem de dados.

7. Digite os valores apropriados para o **Nome do Servidor**, **Número da Porta** e o **Nome do Banco de Dados** (o nome deve ser CVSAMPLE).

**Nota:** Se você precisar de ajuda para determinar os valores corretos para estes campos, entre em contato com o administrador do banco de dados.

8. Digite um **Nome de Usuário Padrão** e uma **Senha Padrão**.

O nome do usuário e a senha devem ser válidos no banco de dados relacional. O nome do usuário e a senha padrão são sempre utilizados quando um cubo do DB2 Alphablox acessa o banco de dados relacional. O usuário do banco de dados que você utiliza deve ter direitos de acesso de leitura no banco de dados.

**Nota:** O valor de **Utilizar Nome do Usuário e Senha do DB2 Alphablox** é ignorado quando a origem de dados relacional especificada está sendo utilizada para ocupar um cubo do DB2 Alphablox. As ACLs (Access Control Lists) podem ser utilizadas para restringir o acesso aos cubos do DB2 Alphablox. Para obter informações sobre ACLs, consulte o *Administrator's Guide*.

9. Os valores de **Número Máximo de Linhas** e **Número Máximo de Colunas** são ignorados quando a origem de dados está sendo utilizada para preencher um cubo do DB2 Alphablox. Você ainda podem digitar valores e eles serão utilizados quando outros aplicativos utilizarem a origem de dados.
10. Configure o valor de **Rastreamento de JDBC Ativado** como **Não**, a não ser que você queira gravar informações de registro do JDBC no arquivo de registro do DB2 Alphablox. Ative o rastreamento do JDBC apenas se você tiver encontrando problemas e precisar depurar suas causas.
11. Clique no botão **Salvar** para salvar a definição da origem de dados.

Você definiu agora a origem de dados do DB2 Alphablox para CVSAMPLE. Agora é possível criar uma definição de cubo do DB2 Alphablox para acessar os metadados de cubos de Cube Views nesta origem de dados do DB2.

---

## Definindo uma Origem de Dados do Alphablox Cube Server Adaptor

Nesta tarefa, você definirá uma origem de dados do DB2 Alphablox que utiliza o Alphablox Cube Server Adapter.

Pré-requisitos: Crie a definição da origem de dados do DB2 Alphablox para o banco de dados CVSAMPLE do DB2.

Para definir uma origem de dados do DB2 Alphablox Cube Server Adapter:

1. Efetue logon nas Páginas de Administração do DB2 Alphablox como o usuário admin (ou como um usuário com direitos de administrador).
2. Clique na guia **Administração**.
3. Clique no link **Origem de Dados**.
4. Clique no botão **Criar**.
5. No menu **Adaptador**, selecione a opção **Alphablox Cube Server Adapter**.
6. No campo **Nome da Origem de Dados**, digite DB2AlphabloxCubes como o nome a ser utilizado para sua origem de dados.
7. Clique no botão **Salvar** para salvar a definição da origem de dados.

Você definiu uma origem de dados do DB2 Alphablox que pode ser utilizada para acessar cubos do DB2 Alphablox. A seguir você precisa definir um cubo do DB2 Alphablox que você pode acessar.

---

## Definindo um Cubo do DB2 Alphablox

Nesta tarefa, você definirá um cubo do DB2 Alphablox baseado nos metadados de um cubo CVSAMPLE do DB2 Cube Views.

**Pré-requisitos:** Defina a origem de dados relacional do DB2. Defina uma origem de dados do DB2 Alphablox Cube Server Adapter.

Para definir as propriedades gerais de um cubo DB2 Alphablox:

1. Efetue logon nas Páginas de Administração do DB2 Alphablox como o usuário admin (ou como um usuário com direitos de administrador).
2. Clique na guia **Administração**.
3. Clique no link **Cubos**.
4. Clique no botão **Criar**. A janela Administração do Cubo é aberta.
5. Defina o novo cubo:
  - a. No campo **Nome do Cubo do DB2 Alphablox**, digite **CVSales**.
  - b. Selecione a opção **Ativado**, localizada próxima do campo **Nome do Cubo do DB2 Alphablox**. A seleção desta opção resulta na inicialização automática do cubo quando o servidor é reiniciado.
  - c. No menu **Origem de Dados Relacionais**, selecione **MyDB2**, a origem de dados relacionais que você criou para este tutorial.
  - d. Deixe a opção **Função de Segurança** desmarcada. Esta opção pode ser utilizada para limitar os usuários que podem acessar um cubo específico.
6. Ative as configurações do DB2 Cube Views e especifique os metadados a serem utilizados:
  - a. Selecione a opção **Ativar Configurações do DB2 Cube Views**.
  - b. No menu **Modelo de Cubo**, selecione **CVSAMPLE.Sales**.
  - c. No menu **Cubo**, selecione o cubo **Vendas Gerais**.
  - d. Selecione o botão de rádio **Utilizar Nomes de Negócios** para especificar os nomes. A seleção desta opção utiliza os nomes de membros que normalmente são mais significativos e legíveis.
  - e. Clique no botão **Importar Definição de Cubo**. O uso desta opção permite importar uma definição de cubo e pré-preencher medidas e dimensões em seu cubo do DB2 Alphablox. A definição de cubo importada reflete o cubo do DB2 Cube Views da maneira mais aproximada possível que o DB2 Alphablox pode corresponder, com base no suporte de metadados do DB2 Cube Views. Quando você estiver mais experiente no trabalho com cubos, poderá modificar medidas e dimensões para atender às suas necessidades.
  - f. Clique no botão **Mostrar Registro de Importação** para ver um registro que especifica informações e mensagens de depuração relacionadas à operação de importação. Para este tutorial, esta etapa é incluída apenas para ajudá-lo a familiarizar-se com este recurso.
  - g. Selecione a opção **Importar Definição de Cubo na Inicialização, Reconstrução e Edição**. Esta opção fará com que o seu cubo do DB2 Alphablox utilize a última definição do DB2 Cube Views a cada vez que o cubo do DB2 Alphablox for iniciado, reconstruído ou aberto para edição. Mais familiarizado com o DB2 Alphablox e o DB2 Cube Views, você pode importar a definição de cubo e customizá-la.

7. Clique no botão **Salvar** para salvar a definição do cubo do DB2 Alphablox.

Você criou uma definição de cubo do DB2 Alphablox. Agora você pode iniciar o novo cubo CVSales.

---

## Iniciando o Cubo do seu DB2 Alphablox

Nesta tarefa, você inicia o cubo CVSales utilizando as Páginas de Administração do DB2 Alphablox.

**Pré-requisitos:** Defina a origem de dados relacional do DB2. Defina uma origem de dados do DB2 Alphablox Cube Server Adapter. Defina o cubo do DB2 Alphablox.

Para iniciar o cubo do CVSales:

1. Efetue logon nas Páginas de Administração do DB2 Alphablox como o usuário admin (ou como um usuário com direitos de administrador).
2. Clique na guia **Administração**.
3. Na seção **Gerenciamento do Tempo de Execução**, clique no link **Cubos**.
4. Na lista **Cubos do DB2 Alphablox**, selecione o cubo DB2 Alphablox que deseja iniciar.
5. Clique no botão **Iniciar**. Quando o cubo tiver sido iniciado, o campo de status exibirá **Executando**.

Você tem agora um cubo de amostra do DB2 Alphablox em execução. Você pode começar a construção de aplicativos utilizando o cubo do DB2 Alphablox que você acabou de criar. Você pode executar uma verificação rápida no novo cubo utilizando o aplicativo Query Builder, selecionando o cubo como a origem de dados e executando consultas MDX contra o cubo.

---

## Avisos

Estas informações foram desenvolvidas para produtos e serviços oferecidos nos Estados Unidos.

É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos nesta publicação em outros países. Consulte um representante IBM local para obter informações sobre produtos e serviços disponíveis atualmente em sua área. Qualquer referência a produtos, programas ou serviços IBM não significa que apenas produtos, programas ou serviços IBM possam ser utilizados. Qualquer produto, programa ou serviço funcionalmente equivalente, que não infrinja nenhum direito de propriedade intelectual da IBM, poderá ser utilizado em substituição a este produto, programa ou serviço. Entretanto, a avaliação e verificação da operação de qualquer produto, programa ou serviço não-IBM são de responsabilidade do Cliente.

A IBM pode ter patentes ou solicitações de patentes pendentes relativas a assuntos tratados nesta publicação. O fornecimento desta publicação não garante ao Cliente nenhum direito sobre tais patentes. Pedidos de licença devem ser enviados, por escrito, para:

*Gerência de Relações Comerciais e Industriais da IBM Brasil, Av. Pasteur, 138-146, Botafogo, Rio de Janeiro, RJ, CEP 22290-240.*

Para pedidos de licença relacionados a informações de DBCS (Conjunto de Caracteres de Byte Duplo), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie pedidos de licença, por escrito, para:

*IBM World Trade Asia Corporation, Licensing, 2-31 Roppongi 3-chome, Minato-ku, Tokyo 106-0032, Japan*

*O parágrafo a seguir não se aplica a nenhum país em que tais disposições não estejam de acordo com a legislação local: A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO "NO ESTADO EM QUE SE ENCONTRA", SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS NÃO SE LIMITANDO ÀS GARANTIAS IMPLÍCITAS DE NÃO-INFRAÇÃO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns países não permitem a exclusão de garantias expressas ou implícitas em certas transações; portanto, essa disposição pode não se aplicar ao Cliente.*

Esta publicação pode conter imprecisões técnicas ou erros tipográficos. Periodicamente, são feitas alterações nas informações aqui contidas; tais alterações serão incorporadas em futuras edições desta publicação. A IBM pode, a qualquer momento, aprimorar e/ou alterar os produtos e/ou programas descritos nesta publicação, sem aviso prévio.

Referências nestas informações a Web sites não-IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a esses Web sites. Os materiais contidos nesses Web sites não fazem parte dos materiais desse produto IBM e a utilização desses Web sites é de inteira responsabilidade do Cliente.

A IBM pode utilizar ou distribuir as informações fornecidas da forma que julgar apropriada sem incorrer em qualquer obrigação para com o Cliente.

Licenciados deste programa que desejam obter informações sobre este assunto com objetivo de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) a utilização mútua das informações trocadas, devem entrar em contato com:

*Gerência de Relações Comerciais e Industriais da IBM Brasil. Av. Pasteur, 138-146, Botafogo, Rio de Janeiro, RJ, CEP 22290-240.*

Tais informações podem estar disponíveis, sujeitas a termos e condições apropriadas, incluindo em alguns casos o pagamento de uma taxa.

O programa licenciado descrito neste documento e todo material licenciado disponível são fornecidos pela IBM sob os termos do Contrato com o Cliente IBM, do Contrato de Licença de Programas Internacional IBM ou de qualquer outro contrato equivalente.

Todos os dados de desempenho aqui contidos foram determinados em um ambiente controlado. Portanto, os resultados obtidos em outros ambientes operacionais podem variar significativamente. Algumas medidas podem ter sido tomadas em sistemas de nível de desenvolvimento e não há garantia de que estas medidas serão iguais em sistemas geralmente disponíveis. Além disso, algumas medidas podem ter sido estimadas por extrapolação. Os resultados reais podem variar. Usuários deste documento devem verificar os dados aplicáveis para seu ambiente específico.

As informações relativas a produtos não-IBM foram obtidas junto aos fornecedores dos respectivos produtos, de seus anúncios publicados ou de outras fontes disponíveis publicamente. A IBM não testou estes produtos e não pode confirmar a precisão de seu desempenho, compatibilidade nem qualquer outra reivindicação relacionada a produtos não-IBM. Dúvidas sobre os recursos de produtos não-IBM devem ser encaminhadas diretamente a seus fornecedores.

Todas as declarações relacionadas aos objetivos e intenções futuras da IBM estão sujeitas a alterações ou cancelamento sem aviso prévio e representam apenas metas e objetivos.

Estas informações contêm exemplos de dados e relatórios utilizados nas operações diárias de negócios. Para ilustrá-los da forma mais completa possível, os exemplos podem incluir nomes de indivíduos, empresas, marcas e produtos. Todos estes nomes são fictícios e qualquer semelhança com nomes e endereços utilizados por uma empresa real é mera coincidência.

Estas informações contêm programas de aplicativos de exemplo na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. O Cliente pode copiar, modificar e distribuir estes programas de exemplo sem a necessidade de pagar à IBM, com objetivos de desenvolvimento, utilização, marketing ou distribuição de programas aplicativos em conformidade com a interface de programação de aplicativo para a plataforma operacional para a qual os programas de exemplo são criados. Esses exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar a confiabilidade, manutenção ou função destes programas. O Cliente pode copiar, modificar e distribuir estes programas de exemplo de qualquer maneira

sem pagamento à IBM, com objetivos de desenvolvimento, utilização, marketing ou distribuição de programas aplicativos em conformidade com interfaces de programação de aplicativos da IBM.

---

## Marcas Registradas

Os termos a seguir são marcas ou marcas registradas da International Business Machines Corporation nos Estados Unidos e/ou em outros países:

Cube Views	DB2	DB2 OLAP Server
DB2 Universal Database IBM	Informix	Rational
WebSphere		

Alphablox e Blox são marcas ou marcas registradas da Alphablox Corporation nos Estados Unidos e/ou em outros países.

Java e todas as marcas registradas baseadas em Java são marcas registradas da Sun Microsystems, Inc. nos Estados Unidos e/ou em outros países.

Linux é marca registrada de Linus Torvalds nos Estados Unidos e/ou em outros países.

Outros nomes de empresas, produtos ou serviços podem ser marcas registradas ou marcas de serviço de terceiros.





---

# Índice Remissivo

## D

- dados de amostra
  - criando a definição de cubo QCC, repositório baseado em arquivos 7
  - criando a definição de cubo QCC, repositório baseado em banco de dados 7
- definido a origem de dados da QCC 13
- instalando
  - DB2 2
  - Derby 3
  - Hyperion Essbase 6.5 e DB2 OLAP Server 8.1 8
  - Hyperion Essbase 7.1 e DB2 OLAP Server 8.2 9
  - Informix 3
  - Microsoft SQL Server 4
  - Microsoft SQL Server 2000 Analysis Services 11
  - Microsoft SQL Server 2005 Analysis Services 12
  - Oracle 5
  - Sybase 5
- instalando as tabelas da QCC 2
- multidimensional 8
- visão geral dos dados de amostra da QCC 1

## P

- portlet
  - amostra 25
  - estrutura do JSP 26

## I

- instalação
  - dados de amostra
    - criando a definição de cubo QCC, repositório baseado em arquivos 7
    - criando a definição de cubo QCC, repositório baseado em banco de dados 7
  - DB2 2
  - definido a origem de dados da QCC 13
  - Derby 3
  - Hyperion Essbase 6.5 e DB2 OLAP Server 8.1 8
  - Hyperion Essbase 7.1 e DB2 OLAP Server 8.2 9
  - Informix 3
  - instalando as tabelas da QCC 2
  - Microsoft SQL Server 4
  - Microsoft SQL Server 2000 Analysis Services 11
  - Microsoft SQL Server 2005 Analysis Services 12
  - multidimensional 8
  - Oracle 5
  - Sybase 5







Número do Programa: 5724-L14

Impresso em Brazil

G517-8371-02

