

IBM InfoSphere Federation Server

Versão 10.1

*Guia de Configuração para Origens de
Dados Federados*



IBM InfoSphere Federation Server

Versão 10.1

*Guia de Configuração para Origens de
Dados Federados*



Nota

Antes de utilizar estas informações e o produto que elas suportam, leia as informações contidas em “Avisos e Marcas Registradas” na página 389.

Índice

Capítulo 1. Configurando o Servidor

Federado 1

Executando a Ferramenta de Configuração de Wrapper	1
Objetos Federados no IBM Data Studio	2
Criando Objetos Federados: Administration Explorer	3
Criando e Gerenciando Objetos Federados: Editor de SQL e XQuery	8
Emitindo Comandos em um Sistema Federado	10
Verificando a configuração do servidor federado	10
Ativando o Servidor Federado para Acessar as Origens de Dados	10
Configurando Variáveis de Ambiente Necessárias para Clientes da Origem de Dados	11
Verificando os Arquivos de Biblioteca que estão Vinculados ao Software Cliente da Origem de Dados	13
Gerenciando a chave de licença do produto e a política	16
Configurando a Federação para o Recurso DB2 PureScale	16
Criando um Banco de Dados Federado	17
Conjuntos de Códigos, Sequências de Intercalação e Suporte ao Idioma Nacional	17
Configurando o Servidor Federado para Acessar Origens de Dados	22
Distinção entre maiúsculas e minúsculas e uso correto de aspas	25
Configurando Vários Servidores Federados para Acessar Origens de dados	28

Capítulo 2. Configurando Origens de Dados 29

Configurando o Acesso às Origens de Dados do BioRS	29
Wrapper BioRS	29
Incluindo Origens de Dados BioRS em um Servidor Federado	30
Funções Customizadas e Consultas BioRS	45
Otimizando o Desempenho do Wrapper BioRS	56
Configurando o Acesso às Origens de Dados do DB2	62
Catalogando uma Entrada de Nó do DB2	62
Catalogando o Banco de Dados DB2 Remoto	62
Registrando o Wrapper DB2	63
Registrando as Definições do Servidor para Origens de Dados DB2	64
Criando Mapeamentos de Usuário para Origens de Dados do DB2	66
Testando a Conexão com o Servidor da Origem de Dados do DB2	68
Registrando Apelidos para Tabelas e Visualizações do DB2	69

Configurando o Acesso às Origens de Dados do Excel	71
Wrapper do Excel	71
Incluindo Origens de Dados do Excel em um Servidor Federado	75
Configurando o Acesso a Origens de Dados do Informix	82
Configurando e Testando o Arquivo de Configuração do ClienteInformix	83
Definindo as Variáveis de Ambiente do Informix	84
Registrando o Wrapper do Informix	87
Registrando as Definições do Servidor para uma Origem de Dados do Informix	89
Criando os Mapeamentos de Usuário para uma Origem de Dados do Informix	92
Testando a Conexão com o Servidor Informix	93
Registrando Apelidos para Tabelas, Visualizações e Sinônimos do Informix	95
Configurando acesso a origens de dados JDBC	96
Preparando o Servidor Federado para Acessar Origens de Dados através do JDBC	97
Registrando o Wrapper JDBC	98
Registrando Definições de Servidor para Origens de Dados JDBC	100
Criando Mapeamentos de Usuário para Origens de Dados JDBC	103
Testando Conexões com Servidores de Origem de Dados JDBC	104
Registrando Apelidos para Tabelas e Visualizações da Origem de Dados JDBC	105
Configurando Acesso às Origens de Dados do Microsoft SQL Server	106
Preparando o Servidor Federado para Acessar as Origens de Dados Microsoft SQL Server (Windows)	107
Preparando o Servidor Federado para Acessar Origens de Dados do Microsoft SQL Server (Linux, UNIX)	107
Configurando as Variáveis de Ambiente do Microsoft SQL Server	109
Registrando o Wrapper do Microsoft SQL Server	111
Registrando as Definições do Servidor para uma Origem de Dados do Microsoft SQL Server	112
Criando os Mapeamentos de Usuário para uma Origem de Dados do Microsoft SQL Server	114
Testando a Conexão com o Servidor RemotoMicrosoft SQL Server	116
Registrando Apelidos para as Tabelas e Visualizações do Microsoft SQL Server	117
Utilizando Informações de Rastreamento do ODBC para Resolver Problemas de Conexões com as Origens de Dados do Microsoft SQL Server	118
Configurando o Acesso às Origens de Dados ODBC	119

Preparando o Servidor Federado para Acessar as Origens de Dados por meio do ODBC (Windows)	121	Configurando e Testando o Arquivo de Configuração do Cliente Sybase (UNIX)	181
Preparando o Servidor Federado para Acessar Origens de Dados por meio do ODBC (Linux, UNIX).	121	Registrando o Wrapper do Sybase	182
Registrando o Wrapper ODBC.	123	Registrando Definições de Servidor para uma Origem de Dados Sybase	183
Registrando as Definições de Servidor para uma Origem de Dados ODBC	126	Criando os Mapeamentos de Usuário para uma Origem de Dados Sybase	185
Criando um Mapeamento de Usuário para uma Origem de Dados ODBC	128	Testando a Conexão com o Servidor Sybase	187
Testando a Conexão com o Servidor de Origem de Dados ODBC	129	Registrando Apelidos para Tabelas e Visualizações do Sybase	188
Registrando Apelidos para Tabelas e Visualizações de Origens de Dados ODBC.	131	Resolução de Problemas da Configuração do Wrapper do Sybase	189
Otimizando o Desempenho do Wrapper ODBC com o Utilitário de Ajuste do ODBC (db2fedsvrcfg)	132	Configurando o Acesso às Origens de Dados do Arquivo Estruturado em Tabela	191
Acessando Dados do Excel Utilizando o Wrapper ODBC	135	Arquivos Estruturados em Tabela - Visão Geral	191
Configurando Acesso ODBC para Origens de Dados do IBM InfoSphere Classic Federation Server para z/OS	139	Atributos para Arquivos Estruturados em Tabela	192
Configurando o Acesso ODBC às Origens de Dados Netezza	141	Wrapper de Arquivos Estruturados em Tabela	192
Configurando o Acesso às Origens de Dados do DB OLE	142	Incluindo Origens de Dados de Arquivo com Estrutura de Tabela em um Servidor Federado	193
Registrando o Wrapper OLE DB	143	Modelo de Controle de Acesso de Arquivo para o Wrapper de Arquivo Estruturado em Tabela	201
Registrando as Definições de Servidor para uma Origem de Dados OLE DB	144	Orientações para Otimizar o Desempenho de Consultas para o Wrapper do Arquivo Estruturado em Tabela	202
Criando os Mapeamentos de Usuário para uma Origem de Dados OLE DB	145	Configurando o Acesso às Origens de Dados Teradata	202
Configurando o Acesso às Origens de Dados do Oracle	146	Testando a Conexão com o Servidor Teradata	203
Definindo as Variáveis de Ambiente do Oracle	147	Verificando se a Biblioteca do Teradata Está Ativada para Link de Tempo de Execução (AIX).	203
Configurar e Testar o Arquivo de Configuração do Cliente Oracle	151	Definindo as Variáveis de Ambiente do Teradata	204
Registrando o Wrapper Oracle.	152	Registrando o Wrapper Teradata	209
Registrando Definições de Servidor para uma Origem de Dados Oracle	153	Registrando as Definições de Servidor para uma Origem de Dados Teradata	210
Criando Mapeamentos de Usuários para uma Origem de Dados Oracle	155	Criando o Mapeamento de Usuário para uma Origem de Dados Teradata	212
Testando a Conexão com o Servidor Oracle	157	Testando a Conexão com o Servidor Teradata	214
Registrando Pseudônimos para Tabelas e Visualizações do Oracle	158	Registrando Apelidos para as Tabelas e Visualizações do Teradata	215
Configurando o Acesso aos Scripts como Origens de Dados.	160	Resolução de Problemas da Configuração da Origem de Dados do Teradata.	217
Visão Geral do Wrapper de Script	160	Configurando o Acesso às Origens de Dados de Serviços da Web	219
Incluindo Scripts como Origens de Dados em um Sistema Federado	162	Serviços da Web e o Wrapper de Serviços da Web	219
Registrando apelidos para scripts.	169	Registrando o Wrapper de Serviços da Web	226
Consultas SQL com o Wrapper de Script	173	Registrando a Definição de Servidor para Origens de Dados de Serviços da Web	227
Otimizando o Desempenho do Wrapper de Script	175	Ativando a Segurança através do Wrapper de Serviços da Web	229
Configurando o Acesso às Origens de Dados do Sybase.	175	Registrando Apelidos para as Origens de Dados de Serviços da Web	230
Suporte do Wrapper do Sybase para ASE (Adaptive Server Enterprise)	176	Restrições de consulta para Wrappers de Serviços da Web	241
Definindo as Variáveis de Ambiente do Sybase	177	Origens de Dados dos Serviços da Web - Exemplo de Consultas	244
Configurando e Testando o Arquivo de Configuração do Cliente Sybase (Windows)	180	Configurando o Acesso às Origens de Dados do XML	247
		wrapper do XML	247
		Incluindo XML em um Sistema Federado	250
		Consultas para Origens de Dados XML.	266

Capítulo 3. Suporte de origem de dados para recursos federados . . . 273

Capítulo 4. Referência de Opções de Origens de Dados . . . 277

Referência de Opções do BioRS 277
Referência das Opções do Banco de Dados DB2 281
Referência de Opções do Excel 289
Referência das Opções do Informix 290
Referência de Opções JDBC 295
Referência de Opções do Microsoft SQL Server 302
Referência de Opções ODBC 307
Referência de Opções Oracle 313
Referência de Opções de Script 318
Referência de Opções Sybase 323
Referência de Opções Teradata 329
Referência das Opções de Arquivos Estruturados em Tabelas 333
Referência de Opções de Serviços da Web 335
Referência de Opções XML 342

Capítulo 5. Visualizações na Tabela de Catálogos Globais Contendo Informações sobre Federação . . . 349

Capítulo 6. Opções de Mapeamento de Funções para Sistemas Federados. 353

Capítulo 7. Tipos de servidores válidos em instruções SQL 355

Capítulo 8. Mapeamentos de Tipos de Dados 357

Mapeamentos de tipo de dados de encaminhamento padrão 357
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de dados do Banco de Dados DB2 para Linux, UNIX e Windows 357
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do DB2 para System i 358
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do DB2 para VM e VSE 359
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do DB2 para z/OS 360
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Informix 360
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do JDBC 361
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Microsoft SQL Server 363

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do ODBC 364
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Oracle NET8 365
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Sybase 366
Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Teradata 368
Mapeamentos de Tipos de Dados de Encaminhamento de Amostra 369
Mapeamentos de tipo de dados reversos padrão 371
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Banco de Dados DB2 para Linux, UNIX, e Windows 372
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do DB2 para System i 372
Mapeamentos de Tipos de Dados Reversos Padrão para origens de dados do DB2 para VM e VSE 373
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do DB2 para z/OS 374
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Informix 374
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do JDBC 375
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Microsoft SQL Server 376
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do ODBC 377
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Oracle NET8 377
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Sybase 378
Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Teradata 379
Mapeamentos de Tipos Padrão do Unicode 380
Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do JDBC 380
Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do JDBC 380
Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do Microsoft SQL Server 380
Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do Microsoft SQL Server 381
Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do NET8 381
Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do NET8 382

Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do ODBC	382
Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do ODBC	383
Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do Sybase	383
Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do Sybase.	383

Tipos de Dados Suportados para Origens de Dados Não Relacionais	384
---	-----

Documentação Acessível 387

Avisos e Marcas Registradas 389

Índice Remissivo 393

Capítulo 1. Configurando o Servidor Federado

Antes de configurar as origens de dados, configure o servidor federado e verifique se o mesmo está configurado adequadamente.

Procedimento

Para configurar o servidor federado, conclua estas tarefas:

Executando a Ferramenta de Configuração de Wrapper

A ferramenta de configuração de wrapper do Federation Server é apresentada na Versão 10. A ferramenta de configuração fornece uma interface de GUI para configurar wrappers relacionais.

Antes de Iniciar

Antes de instalar e usar a ferramenta de configuração, certifique-se de que o sistema federado esteja configurado corretamente. Os wrappers devem estar instalados para as origens de dados que deseja configurar.

Sobre Esta Tarefa

A ferramenta de configuração detecta qualquer instância do DB2 existente. Selecione qual instância e wrapper você deseja configurar e insira as variáveis de ambiente necessárias e opcionais para cada wrapper. Com base em sua entrada, a ferramenta de configuração grava variáveis de ambiente no arquivo db2dj.ini e exporta o .profile da instância do DB2. Para wrappers do Informix e Teradata, a ferramenta executa djxLink. Após a configuração estar completa, é possível escolher reiniciar o DB2 e ativar a configuração.

Ao iniciar a ferramenta de configuração, o instalador de federação determina o sistema operacional, a autorização do usuário e a versão do DB2 instalada. Com base nessas informações, a ferramenta de configuração apresenta os wrappers que é possível configurar.

Restrições: A ferramenta de configuração suporta os sistemas operacionais Linux e UNIX. É possível usar a ferramenta de configuração para configurar os seguintes wrappers relacionais:

- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

A ferramenta de configuração não suporta os wrappers ODBC e JDBC.

Procedimento

1. Ative a ferramenta de configuração.

Acesse a pasta de compartimento no caminho da instalação /installation_path/bin e execute rwcfg.bin para ativar a ferramenta de

configuração. O ID do usuário deve ter propriedade de administrador se um usuário raiz instalou o programa. O ID do usuário deve ter autoridade de usuário não raiz se um usuário não raiz instalou o produto.

2. Selecione a instância e os wrappers relacionais para configuração.
O painel de Boas-vindas solicita a seleção de uma instância para configuração. No painel Selecionar instância e wrapper para configuração:
 - Selecione uma instância para configuração. O painel exibe os wrappers que foram instalados com base na instância selecionada.
 - Selecione o wrapper que deseja configurar
3. Configure os wrappers relacionais.
Para cada wrapper, será solicitada a entrada de parâmetros necessários e opcionais. A ferramenta de configuração o guia pela configuração de todas as variáveis do sistema e a execução do script djxlink.
4. Revise o resumo de pré-configuração e execute a configuração.
Um painel de resumo de pré-configuração resume sua entrada. É possível retornar a um painel anterior se for necessário fazer mudanças ou continuar a configuração dos componentes selecionados. Ao executar a configuração, a ferramenta de configuração define as variáveis de ambiente de entrada para db2dj.ini e .profile de instância e executa .profile e djxlink se necessário.
5. Reinicie o servidor federado.
Após concluir a configuração, o painel Reiniciar o Servidor Federado é exibido. É possível reiniciar o servidor federado ou esperar para reiniciar mais tarde.
6. Revise o resumo pós-configuração. Após concluir a configuração, o painel Resumo da Configuração é exibido. O painel lista as seguintes informações:
 - O status do componente selecionado.
 - Os detalhes do status de configuração, incluindo os códigos de retorno, as mensagens de aviso e de erro e as informações do arquivo de log.

Objetos Federados no IBM Data Studio

Após instalar e configurar um servidor federado e criar um banco de dados federado, é possível usar o IBM® Data Studio para criar e gerenciar objetos federados.

Usando o Administration Explorer e a Lista de Objetos, é possível configurar um servidor federado para acessar origens de dados suportadas em um servidor de dados remoto.

O suporte do Data Studio para o IBM InfoSphere Federation Server é limitado a recursos específicos do Data Studio. Esses recursos que podem ser usados dependem da origem de dados que deseja acessar. As limitações são resumidas na lista a seguir:

- O Data Studio possui duas visualizações que podem ser usadas para navegar para objetos do banco de dados: Administration Explorer e Data Source Explorer. Você deve usar o Administration Explorer. O Data Source Explorer não suporta objetos federados.
- O suporte do Data Studio para criação e gerenciamento de objetos federados do Administration Explorer é limitado às seguintes origens de dados:
 - DB2 Database para Linux, UNIX e Windows
 - DB2 para i
 - DB2 para z/OS

- Oracle
- Para outras origens de dados, é possível usar o Editor de SQL e XQuery no Data Studio para emitir e implementar comandos como alternativa à emissão de instruções SQL a partir da linha de comandos do DB2.

Use os seguintes tópicos para se familiarizar com os recursos do Data Studio que se aplicam ao tipo de origem de dados com a qual está trabalhando.

- Para obter informações sobre a configuração de acesso às origens de dados DB2 suportadas e a origem de dados Oracle, consulte “Criando Objetos Federados: Administration Explorer”.
- Para obter informações sobre a configuração de acesso a outras origens de dados, consulte “Criando e Gerenciando Objetos Federados: Editor de SQL e XQuery” na página 8.

Objetos Federados no Roteiro do Data Studio

O roteiro a seguir lista os recursos de instalação, configuração e administração para trabalhar com objetos federados no Data Studio.

Tabela 1. Roteiro para o Gerenciamento de Objetos Federados no Data Studio

Tarefa	Recurso
Instalar o Data Studio	Você deve instalar o cliente completo do Data Studio ou o cliente de administração do Data Studio. Como opção, é possível instalar o console da web do Data Studio se optar por usar o planejamento de tarefas para a execução de scripts SQL.
Criar uma conexão	Conectando-se a bancos de dados
Exibir e gerenciar objetos	Exibindo e gerenciando objetos de banco de dados na Lista de Objetos
Gerenciar mudanças no objeto	Gerenciando mudanças no objeto de banco de dados usando a visualização Propriedades
Editar comandos	Editando comandos manualmente com o editor de SQL e XQuery
Gerenciar privilégios	Gerenciamento de Privilégios
Trabalhar com planos de mudança	Cenário: Fazendo mudanças básicas no objeto de banco de dados com um plano de mudança
Administrar bancos de dados	Administrando bancos de dados com assistentes de tarefa

Informações detalhadas sobre todos os recursos do Data Studio encontram-se no centro de informações do IBM Data Studio Versão 3.1.1.

Criando Objetos Federados: Administration Explorer

Para as origens de dados DB2 suportadas e a origem de dados Oracle, é possível usar o Data Studio Administration Explorer para configurar um servidor federado para acessar origens de dados em um servidor de dados remoto.

É possível configurar seu sistema federado criando primeiro os objetos federados individuais necessários para o acesso a uma origem de dados. Depois, é possível trabalhar com esses objetos no Administration Explorer para gerenciar os objetos

individualmente ou para configurar diversos objetos federados e executar tarefas adicionais de configuração e administração.

Criando Objetos Federados Individuais

É possível usar o Administration Explorer do Data Studio para criar cada um dos objetos federados requeridos para configurar um servidor federado para acessar origens de dados suportadas do DB2 e a origem de dados do Oracle.

Sobre Esta Tarefa

Este procedimento explica como usar o Administration Explorer para criar uma conexão com o banco de dados e, em seguida, criar e implementar objetos federados requeridos para acessar as origens de dados suportadas do DB2 e a origem de dados do Oracle.

O Administration Explorer é uma árvore de objetos que contém objetos do banco de dados. Os objetos são agrupados em pastas com base no tipo de objeto. Para criar e gerenciar objetos federados, trabalhe com a pasta **Objetos Federados do Banco de Dados** e as outras áreas principais do Administration Explorer:

Lista de Objetos

É possível usar a área da lista de objetos para exibir ou editar o conteúdo de um objeto.

Visualização de propriedades

A visualização Propriedades é onde as propriedades do objeto são exibidas e onde se pode criar e modificar definições do objeto.

Procedimento

1. Conecte a um banco de dados usando um dos seguintes métodos.
 - Crie uma nova conexão com o banco de dados.
 - a. No Administration Explorer, clique em **Novo > Nova Conexão com um banco de dados**.
 - b. Configure a conexão inserindo parâmetros no diálogo de conexão.
 - c. Clique com o botão direito do mouse para conectar ao banco de dados.
 - Conecte a uma conexão com o banco de dados existente.
 - a. Clique com o botão direito do mouse no perfil de conexão no Administration Explorer.
 - b. Clique em **Conectar**.

Nota: Se um banco de dados DB2 ou federado já estiver catalogado na máquina em que o Data Studio está instalado, os perfis de conexão com o banco de dados serão automaticamente criados quando você ativar o Data Studio.

2. Criar os objetos federados requeridos:
 - Wrappers
 - Definições de servidor
 - Mapeamentos de usuário
 - Nomes alternativos

Também é possível criar procedimentos armazenados federados.

- a. Expanda o conteúdo da pasta **Objetos do Banco de Dados Federado**.
- b. Clique com o botão direito do mouse em uma pasta de objetos federados e selecione um objeto para criar no menu suspenso.

As mudanças feitas em um banco de dados são salvas em um plano de mudança na área de trabalho local. É possível usar o plano de mudança para definir mudanças em seu banco de dados DB2 ou federado e para definir como implementar as mudanças.

Por exemplo, clique com o botão direito do mouse na pasta **Wrappers** e selecione **Criar wrapper**. Ao criar o wrapper, a área Lista de Objetos é aberta e o novo wrapper é exibido na visualização Plano de Mudança Padrão.

- c. Especifique os atributos e opções do objeto na visualização Propriedades.

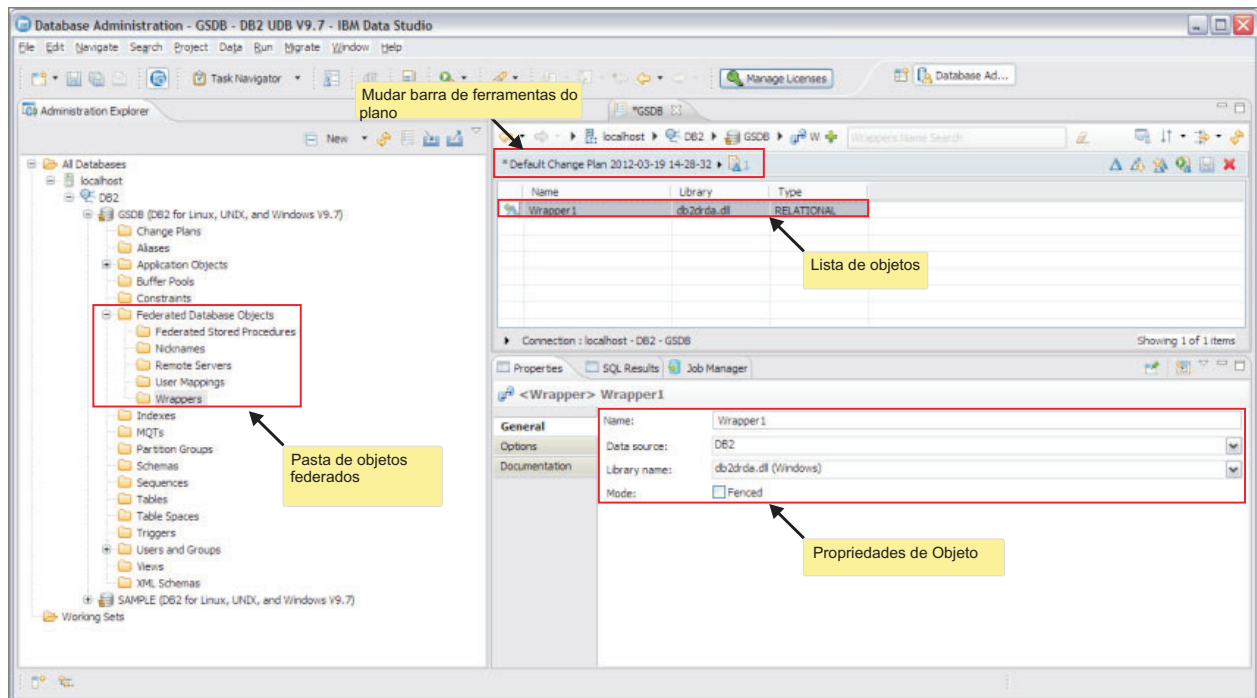


Figura 1. Exemplo do Administration Explorer, lista de objetos e visualização de propriedades.

As propriedades a seguir são requeridas para objetos federados:

Tabela 2. Propriedades do objeto requerido

Objetos federados	Propriedades do objeto requerido
Wrappers	<ul style="list-style-type: none"> • Nome da biblioteca • Opção DBNAME
Servidores	<ul style="list-style-type: none"> • Nome • Tipo • Versão • ID do usuário remoto • Senha remota
Mapeamentos de usuário	<ul style="list-style-type: none"> • ID do usuário local • Nome do servidor • ID do usuário remoto • Senha remota

Tabela 2. Propriedades do objeto requerido (continuação)

Objetos federados	Propriedades do objeto requerido
Nomes alternativos	<ul style="list-style-type: none"> • Nome • Servidor • Esquema remoto • Tabela remota
Procedimentos Armazenados Federados	<ul style="list-style-type: none"> • Nome • Servidor • Esquema remoto • Procedimento remoto

3. Revisar e implementar os comandos.

Quando as definições do objeto estiverem concluídas, clique em **Revisar e Implementar Mudanças** para revisar o DDL gerado e implementar as definições do objeto. É possível editar os comandos, executar os comandos ou planejar para executar posteriormente.

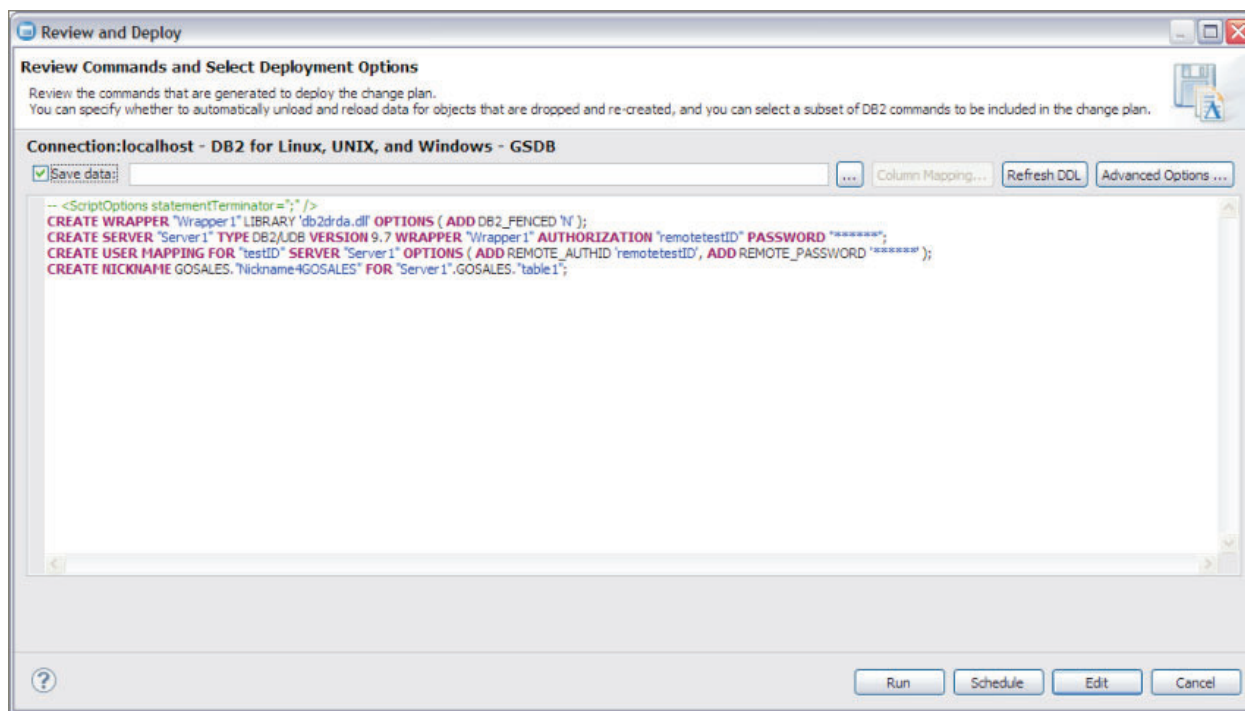


Figura 2. Exemplo do diálogo Revisar e Implementar com definições para um wrapper, servidor, mapeamento de usuários e apelido.

Resultados

Agora é possível acessar objetos federados criados na pasta **Objetos do Banco de Dados Federado** para gerenciar esses objetos. Por exemplo, é possível selecionar um wrapper e escolher **Alterar wrapper** para fazer modificações em uma definição do wrapper ou é possível escolher **Descartar wrapper** para remover um wrapper.

Criando e Gerenciando Diversos Objetos Federados

Após criar objetos federados, é possível trabalhar na visualização Administration Explorer do Data Studio para configurar e gerenciar objetos federados individuais ou diversos objetos federados.

Sobre Esta Tarefa

Para trabalhar com objetos federados de forma mais eficiente, é possível criar ou modificar diversos objetos.

Após criar um objeto, é possível enviar o script para o editor de SQL e XQuery e, em seguida, cortar e colar suas instruções SQL para criar diversos objetos. Por exemplo, é possível criar um wrapper no Administration Explorer. É possível, então, copiar o comando desse wrapper para criar diversos novos wrappers.

O procedimento a seguir descreve como gerenciar diversos objetos.

Procedimento

1. Crie os objetos federados necessários completando as etapas em “Criando Objetos Federados: Administration Explorer” na página 3.
Para obter exemplos detalhados de comandos SQL, consulte Capítulo 2, “Configurando Origens de Dados”, na página 29 para os objetos específicos sendo criados para acessar uma origem de dados.
2. Selecione a opção **Editar** para abrir os comandos para os objetos criados.
Os comandos são exibidos no editor de SQL e XQuery. É possível usar o editor para incluir e editar diversos objetos antes de implementar suas definições de objeto.
3. Copie e edite os comandos para os demais objetos.
Copie comandos de um objeto criado anteriormente. Copie e cole o comando no editor para criar novos objetos.
4. Clique em **Executar SQL** para implementar os scripts SQL.

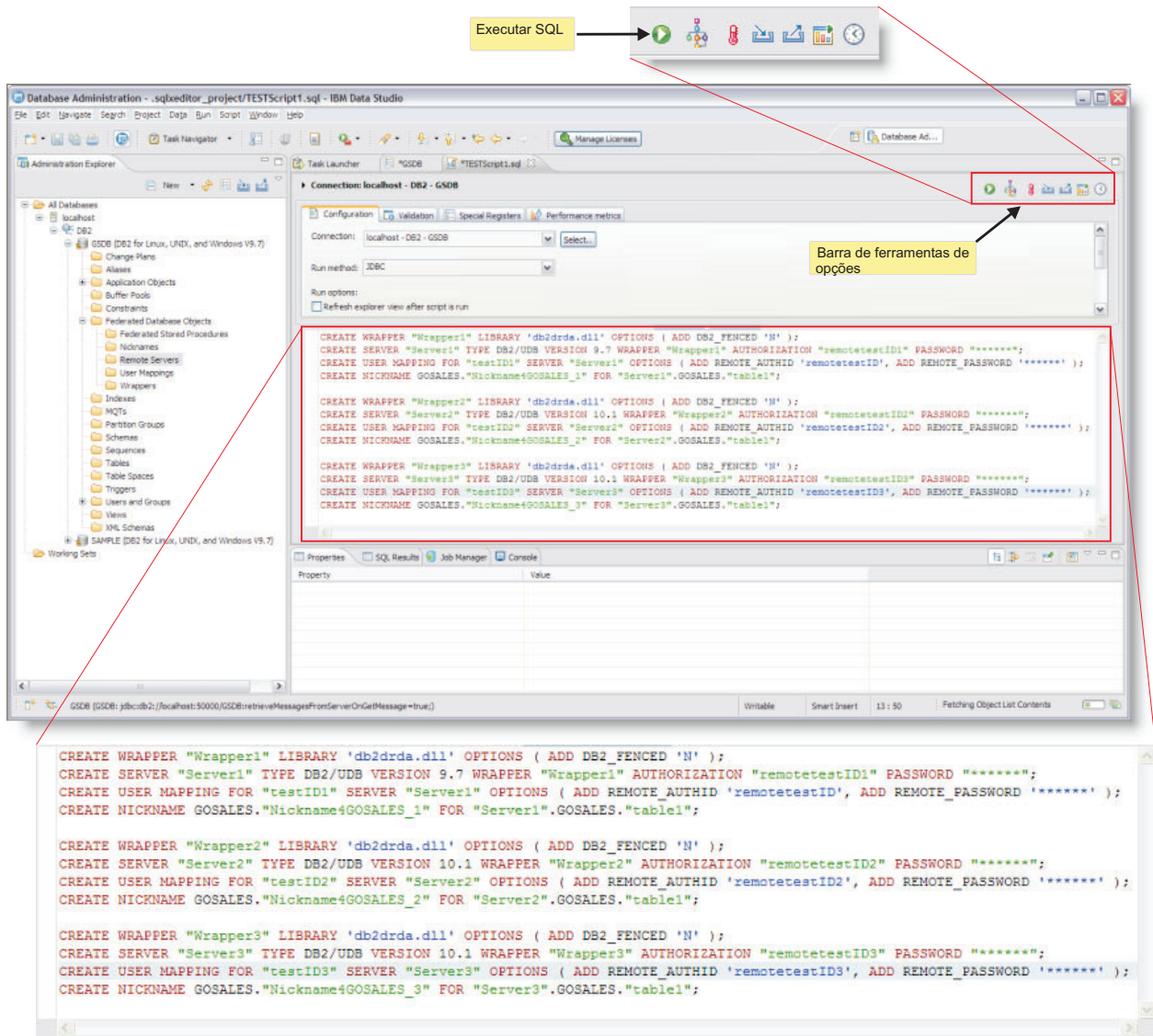


Figura 3. Exemplo de Criação de Diversos Objetos Federados no Editor de SQL e XQuery.

Criando e Gerenciando Objetos Federados: Editor de SQL e XQuery

É possível usar o editor Data Studio SQL e XQuery para emitir e implementar comandos para configurar o acesso a qualquer origem de dados.

Sobre Esta Tarefa

É possível trabalhar com o editor Data Studio SQL e XQuery para criar e editar definições de objetos federados individuais ou múltiplos e implementar comandos. O procedimento a seguir explica como criar uma conexão com o banco de dados e como ativar e usar o editor de SQL e XQuery.

Procedimento

1. Criar os objetos federados requeridos:
 - Wrappers

- Definições de servidor
 - Mapeamentos de usuário
 - Nomes alternativos
- a. Ativar o editor de SQL e XQuery.
No menu **Arquivo**, clique em **Arquivo > Novo > Outro**. Em seguida, selecione **Dados > Script SQL ou XQuery**.
 - b. Conclua o assistente Novo Script SQL ou XQuery e deixe o editor de SQL e de XQuery selecionado.
 - c. Clique em **Concluir**.
A instrução é incluída na pasta **Scripts SQL** do projeto de desenvolvimento de dados selecionada no assistente. A instrução é aberta no editor de SQL e de XQuery.
 - d. Conecte o script a um banco de dados usando o assistente **Selecionar Perfil de Conexão**.
 - e. Conclua as instruções SQL ou XQuery para os objetos que deseja criar.
2. Copie e edite os comandos para os demais objetos.
 3. Clique em **Executar SQL** na barra de ferramentas de opções para executar os scripts SQL.

Exemplos

Os exemplos a seguir demonstram como criar objetos federados usando o editor de SQL e de XQuery.

Exemplo 1: Criar objetos federados para configurar o acesso a diversos servidores para a origem de dados ODBC.

1. Criar cada objeto federado requerido.

```
CREATE WRAPPER ODBC LIBRARY
'libdb2rcodbc.so' OPTIONS (MODULE '/opt/lib/odbc.so');
CREATE SERVER mysql_server TYPE mysql
VERSION 4.0 WRAPPER ODBC
OPTIONS (NODE 'odbc_node', DBNAME 'venice');
CREATE USER MAPPING FOR arturo SERVER mysql_server
OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue');
CREATE NICKNAME cust_europe FOR mysql_server."vinnie"."italy";
```

2. Copie e cole o SQL para criar diversos objetos para os servidores restantes.

```
CREATE WRAPPER ODBC LIBRARY 'libdb2rcodbc.so' OPTIONS (MODULE '/opt/lib/odbc.so');
CREATE SERVER mysql_server TYPE mysql
VERSION 4.0 WRAPPER ODBC
OPTIONS (NODE 'odbc_node', DBNAME 'venice');
CREATE USER MAPPING FOR arturo SERVER mysql_server
OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue');
CREATE NICKNAME cust_europe FOR mysql_server."vinnie"."italy";

CREATE WRAPPER ODBC2 LIBRARY 'libdb2rcodbc.so' OPTIONS (MODULE '/opt/lib/odbc.so');
CREATE SERVER mysql_server2 TYPE mysql
VERSION 4.0 WRAPPER ODBC
OPTIONS (NODE 'odbc_node', DBNAME 'venice');
CREATE USER MAPPING FOR arturo SERVER mysql_server2
OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue');
CREATE NICKNAME cust_europe2 FOR mysql_server2."vinnie"."italy";
```

Exemplo 2: Crie diversas definições do servidor.

1. Crie uma definição do servidor.

```
CREATE SERVER mysql_server2 TYPE mysql
VERSION 4.0 WRAPPER ODBC;
```

2. Copie e cole e edite o DDL para criar definições adicionais do servidor.

```
CREATE SERVER mysql_server2 TYPE mysql
VERSION 4.0 WRAPPER ODBC;
CREATE SERVER mysql2_server2 TYPE mysql
VERSION 4.0 WRAPPER ODBC2;
CREATE SERVER mysql3_server2 TYPE mysql
VERSION 4.0 WRAPPER ODBC3;
```

Emitindo Comandos em um Sistema Federado

Se você for inexperiente em uma federação, revise como iniciar a linha de comandos do DB2. Essa interface é usada para executar muitas tarefas em um sistema federado.

Sobre Esta Tarefa

Para utilizar a linha de comandos do DB2, digite os comandos do DB2 necessários para executar uma tarefa. Se já estiver familiarizado com o DB2, pode ser que prefira utilizar a linha de comandos.

O IBM Data Studio também pode ser usado. Para as origens de dados suportadas, é possível usar o Explorer de Administração. O Explorer de Administração fornece caixas de diálogo e assistentes que orientam na conclusão de uma tarefa, sem que seja necessário conhecer os comandos do DB2 necessários para executar a tarefa. O editor de SQL e XQuery também é uma opção para emitir e implementar comandos como uma alternativa à emissão de instruções SQL a partir da linha de comandos do DB2.

Quando ambas as interfaces estão disponíveis para execução de uma tarefa, a documentação descreve as etapas específicas a serem executadas e os comandos a serem digitados.

Procedimento

Para acessar uma interface, faça o seguinte:

- Para iniciar a linha de comandos do DB2, digite db2 no prompt do sistema operacional. Então é possível digitar os comandos, um após o outro. Ao finalizar a utilização da linha de comandos do DB2, digite quit.
- Para usar o Data Studio, siga as instruções em Objetos federados no Data Studio com base na origem de dados que você precisa acessar.

Verificando a configuração do servidor federado

Quando a federação é instalada, o software tenta configurar o servidor federado. Para evitar problemas ao configurar origens de dados, verifique se o servidor federado está configurado corretamente.

Sobre Esta Tarefa

Para verificar a configuração do servidor federado, conclua estas tarefas:

Ativando o Servidor Federado para Acessar as Origens de Dados

O parâmetro FEDERATED deve ser configurado como YES para ativar o acesso do servidor federado às origens de dados.

Sobre Esta Tarefa

O programa de instalação tenta configurar o parâmetro `FEDERATED`, mas é melhor confirmar se o mesmo está configurado corretamente como `YES`. Se o parâmetro `FEDERATED` não estiver configurado corretamente, todas as tentativas de acesso às origens de dados irão falhar com a mensagem `SQL20076`, código de razão 1.

O procedimento a seguir explica como verificar se o parâmetro `FEDERATED` foi configurado adequadamente e como ativá-lo, se necessário.

Procedimento

1. A partir da linha de comandos do DB2, emita este comando para exibir os parâmetros e suas configurações atuais:
`GET DATABASE MANAGER CONFIGURATION`
2. Verifique o valor do parâmetro `MAX_CONNECTIONS` para determinar se o concentrador está ligado:
 - Se `MAX_CONNECTIONS` for igual a `MAX_COORDAGENTS`, o concentrador está desligado.
 - Se `MAX_CONNECTIONS` for maior do que `MAX_COORDAGENTS`, o concentrador está ligado.

O parâmetro `MAX_CONNECTIONS` não pode estar ativo quando o parâmetro `FEDERATED` está configurado como `YES`. Se o concentrador estiver ligado, altere o valor de `MAX_CONNECTIONS` para que seja igual a `MAX_COORDAGENTS`.

3. Verifique a configuração do parâmetro `FEDERATED`. Se o parâmetro `FEDERATED` estiver configurado como `NO`, altere a configuração para `YES`. Emita o comando a seguir para alterar a configuração:
`UPDATE DATABASE MANAGER CONFIGURATION USING FEDERATED YES`

Configurando Variáveis de Ambiente Necessárias para Clientes da Origem de Dados

Certifique-se de que as variáveis de ambiente necessárias estejam configuradas no arquivo `db2dj.ini`. Esta tarefa é necessária para origens de dados Informix, Microsoft SQL Server, Oracle, Sybase e Teradata, assim como para aplicativos que usam os wrappers JDBC e ODBC.

Antes de Iniciar

O administrador do sistema executa esta tarefa.

Sobre Esta Tarefa

Se o software cliente for instalado antes da instalação do servidor federado, o programa de instalação automaticamente configura as variáveis de ambiente necessárias para as origens de dados selecionadas. Se você instalar o software cliente após instalar o servidor federado, atualizar para uma nova versão do software cliente ou migrar para um novo hardware, será possível usar um dos métodos a seguir para configurar as variáveis de ambiente:

- Use a ferramenta de configuração de wrapper para configurar as variáveis de ambiente.
- Configurar manualmente as variáveis de ambiente.

Esta tabela a seguir lista as variáveis de ambiente necessárias, por origem de dados.

Tabela 3. Variáveis de Ambiente Necessárias, por origem de dados

Origem de Dados	Variável necessária	Valor
Informix	INFORMIXDIR INFORMIXSERVER	Para INFORMIXDIR, o diretório contendo o software cliente Informix. Para INFORMIXSERVER, o nome do servidor padrão Informix.
JDBC	Varia, dependendo do driver JDBC instalado. Se você não especificar o pacote do driver JDBC no parâmetro do servidor DRIVER_PACKAGE da instrução CREATE SERVER, você deve especificar o pacote do driver JDBC na variável de ambiente do sistema "CLASSPATH". Consulte a documentação do seu driver JDBC para obter informações específicas sobre a configuração das variáveis de ambiente do sistema.	Nome do arquivo e diretório da biblioteca do driver JDBC.
Microsoft SQL Server	DJX_ODBC_LIBRARY_PATH	Diretório contendo a biblioteca ODBC.
ODBC	Varia, dependendo do aplicativo ODBC.	Varia, dependendo do aplicativo ODBC.
Oracle	ORACLE_HOME	Diretório contendo o software cliente Oracle.
Sybase	SYBASE, SYBASE_OCS	Para SYBASE, o diretório contendo o software cliente Sybase. Para SYBASE_OCS, o diretório, a versão e o release do software cliente.
Teradata	COPYLIB	O diretório contendo o software cliente Teradata.

Procedimento

1. Abra o arquivo db2dj.ini.

O arquivo db2dj.ini é incluído no servidor federado durante a instalação. Pelo padrão, o arquivo está no local que a variável de registro do BD DB2_DJ_INI especifica. Se a variável de registro DB2_DJ_INI não estiver definida, o arquivo está nestes locais:

- No UNIX, o arquivo está em *instancehome/sql1lib/cfg/db2dj.ini*, onde *instancehome* é o diretório inicial do proprietário da instância
- No Microsoft Windows, o arquivo está em *%DB2PATH%\cfg\db2dj.ini*, onde *%DB2PATH%* é o diretório onde o sistema de banco de dados DB2 está instalado. Por exemplo, C:\Program Files\IBM\sql1lib.

2. Utilize a sintaxe *variable_name=variable_value* para configurar o valor da variável de ambiente. Se *variable_value* for um nome de arquivo ou de diretório, especifique o caminho completo. Por exemplo, para configurar a variável INFORMIDIR para o diretório do Informix no diretório inicial /home/user1, inclua esta entrada no arquivo db2dj.ini:

```
INFORMIXDIR=/home/user1/informix
```

3. A partir da linha de comandos do DB2, emita os seguintes comandos:

```
db2stop  
db2start
```

Pare e, em seguida, reinicie a instância do banco de dados para garantir que as variáveis de ambiente sejam validadas.

4. Se estiver configurando uma sistema com várias partições, copie o arquivo db2dj.ini para cada partição.

Restrições para o Arquivo db2dj.ini

O arquivo db2dj.ini contém as variáveis de ambiente necessárias e opcionais para uma origem de dados.

Quando o arquivo db2dj.ini for modificado, mantenha em mente estas restrições.

Para especificar cada variável, utilize o formato *variable_name=variable_value*

onde

- *variable_name* é o nome da variável de ambiente.
- *variable_value* é o valor.

Para especificar um nome de arquivo ou nome de diretório como *variable_value*, especifique o nome completo. O nome não pode conter metacaracteres de nome de arquivo, como ~ (til), e variáveis de ambiente, como \$HOME. Por exemplo, para configurar a variável de ambiente INFORMIXDIR para o diretório informix e o diretório inicial que é nomeado como /home/user1, especifique este valor no arquivo db2dj.ini:

```
INFORMIXDIR=/home/user1/informix
```

Mantenha em mente estes limites:

- O comprimento máximo para o nome de uma variável de ambiente é de 255 bytes
- O comprimento máximo para o valor de uma variável é de 756 bytes
- O comprimento máximo para uma linha no arquivo db2dj.ini é de 1.021 bytes. Os dados além deste comprimento são ignorados.

Verificando os Arquivos de Biblioteca que estão Vinculados ao Software Cliente da Origem de Dados

Para obter melhores resultados, verifique sempre se os arquivos de biblioteca estão corretamente vinculados ao software cliente da origem de dados antes de utilizar o comando CREATE WRAPPER para registrar o wrapper. Esta tarefa aplica-se somente aos servidores federados que utilizam UNIX e tem o cliente de origem de dados Informix, Microsoft SQL Server, Oracle, Sybase ou Teradata.

Sobre Esta Tarefa

Quando a federação é instalada, é possível configurar os wrappers automaticamente. Se o software cliente da origem de dados for instalado no

servidor federado antes da instalação da federação no servidor e for escolhida a configuração automática do wrapper durante a instalação do programa, os arquivos de biblioteca do wrapper são vinculados ao software cliente da origem de dados correspondente. A vinculação dos arquivos da biblioteca do wrapper depende da origem de dados.

- Origens de dados Informix, Microsoft SQL Server, Sybase e Teradata:
Se o software cliente da origem de dados for instalado depois da instalação da federação, por exemplo para atualizar para uma nova versão do cliente, é necessário vincular manualmente os arquivos de biblioteca do wrapper ao software cliente da origem de dados.
- Origem de dados Oracle:
Os arquivos da biblioteca do wrapper são vinculados dinamicamente. A vinculação dinâmica carregará a biblioteca do cliente Oracle no tempo de execução e analisará automaticamente o endereço da função Oracle Call Interface (OCI).

A seguir são apresentadas causas comuns de falha na vinculação:

- Uma variável de ambiente necessária não está disponível durante a instalação.
- O cliente da origem de dados não está instalado no servidor federado.
- O cliente da origem de dados que está instalado no servidor federado não está em um nível suportado.

Procedimento

Para verificar se os arquivos de biblioteca estão vinculados corretamente:

Procedimento

1. Abra o arquivo de mensagem para a origem de dados. O arquivo de mensagens está no diretório onde o servidor federado está instalado, no subdiretório lib32 ou lib64. Esta tabela lista as origens de dados, arquivos de mensagens e arquivos de bibliotecas.

Tabela 4. Origens de Dados, Arquivos de Mensagens e Arquivos de Bibliotecas

Origem de Dados	Arquivo de mensagens	Arquivos de bibliotecas
Informix	djxlinkInformix.out	AIX - libdb2informix.a HP-UX - libdb2informix.so Linux - libdb2informix.so Solaris - libdb2informix.so
Microsoft SQL Server	djxlinkMssql.out	AIX - libdb2mssql3.a HP-UX - libdb2mssql3.so Linux - libdb2mssql3.so Solaris - libdb2mssql3.so
Sybase	djxlinkSybase.out	AIX - libdb2ctlib.a HP-UX - libdb2ctlib.so Linux - libdb2ctlib.so Solaris - libdb2ctlib.so
Teradata	djxlinkTeradata.out	AIX - libdb2teradata.a HP-UX - libdb2teradata.so Linux - libdb2teradata.so Solaris - libdb2teradata.so

2. Avalie o conteúdo do arquivo de mensagens:
 - Se a vinculação foi bem-sucedida, o arquivo de biblioteca é exibido no diretório onde a federação está instalada, e o arquivo de mensagens exibe uma mensagem de êxito.
 - Se a vinculação falhar, o arquivo de mensagens exibe uma mensagem de erro. Se o arquivo de bibliotecas nem o arquivo de mensagens forem exibidos no diretório onde a federação está instalada, vincule manualmente o arquivo de bibliotecas ao software cliente da origem de dados.

Vinculando Manualmente os Arquivos de Biblioteca ao Software Cliente da Origem de Dados

Se os arquivos de biblioteca não forem exibidos no caminho do diretório, execute um script para vinculá-los ao software cliente da origem de dados.

Antes de Iniciar

- O software cliente da origem de dados deve ser instalado e configurado no servidor federado.
- Pelo padrão, os scripts emitem todas as mensagens em inglês. Para emitir mensagens em um outro idioma, um banco de dados federado deve ser configurado para utilizar a página de códigos para o idioma diferente do inglês.
- As variáveis de ambiente da origem de dados necessárias devem ser definidas.

Tabela 5. Origens de Dados, Scripts e Variáveis de Ambiente Necessárias

Origem de Dados	Script	Variável de ambiente necessária
Informix	djxlinkInformix	INFORMIXDIR
Microsoft SQL Server	djxlinkMssql	DJX_ODBC_LIBRARY_PATH
Sybase	djxlinkSybase	SYBASE, SYBASE_OCS
Teradata	djxlinkTeradata	COPLIB

Procedimento

1. Abra um prompt de comandos doUNIX e digite este comando:


```
cd /opt/IBM/db2/V9.5/bin
```
2. A partir da linha de comandos do DB2, digite os comandos a seguir:


```
db2 disconnect all
db2stop
```
3. Execute o script para cada origem de dados que deseja acessar. Consulte a tabela acima para obter os nomes dos scripts.
4. Se estiver instalando um ambiente com várias partições, repita as Etapas de 1 a 3 para cada instância de banco de dados.
5. Para cada instância de banco de dados, vá para o diretório *install_dir/instance*, onde *install_dir* é o diretório de instalação doDB2, e digite este comando:


```
./db2iupdt instance_name
```

 Este comando atualiza a configuração da instância e dá acesso às origens de dados.
6. Para cada instância de banco de dados, exiba as permissões de arquivo nos arquivos de bibliotecas. Verifique se o proprietário da instância de banco de dados tem permissão para ler e executar os arquivos.

Gerenciando a chave de licença do produto e a política

Cada computador no qual o IBM InfoSphere Federation Server está instalado deve armazenar um arquivo de licença que especifica a chave de licença do produto. A política de licenças controla e monitora o número de usuários que podem se conectar ao servidor federado.

Sobre Esta Tarefa

A chave de licenças está localizada no diretório de licenças do software de instalação. Se a chave de licenças não estiver registrada, registre-a manualmente.

Procedimento

1. Localize seu arquivo de licença no diretório licença do software de instalação do produto:

isfs.lic

é o arquivo de licença do produto integral

isfs_t.lic

é o arquivo de licença do produto Try-and-buy

isfs_d.lic

é o arquivo de licença do Developer Edition

2. Se o arquivo de licença não estiver no diretório, você deve registrar uma chave de licença manualmente, consulte .
3. Especifique a política de licença para controlar e monitorar os usuários que têm permissão para conectar-se ao servidor federado.

Configurando a Federação para o Recurso DB2 PureScale

Para usar funções federadas no ambiente do DB2 pureScale, as bibliotecas do cliente deverão ser instaladas em todos os membros do DB2 pureScale.

Sobre Esta Tarefa

As seguintes restrições aplicam-se ao uso de funções federadas com o Recurso DB2 pureScale:

- As funções de two-phase commit e XML de federação não são suportadas com uma instância pureScale do DB2.
- Para cancelamento de consulta remota, será necessário desativar o novo roteamento automático ininterrupto caso você emita um comando FORCE APPLICATION.
- Para um ACR não ininterrupto, se você cancelar uma instrução federada de longa execução emitindo o comando FORCE APPLICATION, mesmo se SQLCODE -30108 (SQL30108N) for retornado, você não deverá executar essa instrução novamente. A nova execução da instrução poderá resultar em outra interrupção.

Procedimento

Assegure-se de que as bibliotecas do cliente estejam instaladas em todos os membros do DB2 pureScale. A configuração recomendada é instalar todas as bibliotecas do cliente na mesma liberação.

Criando um Banco de Dados Federado

Antes de configurar o servidor federado para acessar as origens de dados, crie um banco de dados federado.

Antes de Iniciar

- Para criar um banco de dados é necessária autoridade SYSADM ou SYSCTRL.
- A federação deve ser instalada em um servidor que atuará como o servidor federado.

Sobre Esta Tarefa

Se as instâncias de banco de dados utilizarem uma configuração de múltiplas partições, todas as partições listadas no arquivo `db2nodes.cfg` serão afetadas na criação do banco de dados. A partição do banco de dados a partir da qual o comando `CREATE DATABASE` é emitido se torna a partição do catálogo para o novo banco de dados.

Procedimento

1. Determine o conjunto de códigos e a seqüência de intercalação que deseja especificar ao criar o banco de dados federado.
2. A partir da linha de comandos do DB2, emita o comando `CREATE DATABASE`.

Por exemplo, para criar um banco de dados nomeado como `federated` que utiliza o conjunto de códigos ISO8859-15 no território do Brasil, emita este comando:

```
CREATE DATABASE
federated
USING CODESET ISO8859-15
TERRITORY BR
```

Conjuntos de Códigos, Sequências de Intercalação e Suporte ao Idioma Nacional

Quando um banco de dados federado é criado, o conjunto de códigos, o território e a seqüência de intercalação são especificados. Estas informações controlam o idioma no qual os dados são armazenados e a seqüência na qual os dados de caracteres são classificados.

Um *conjunto de códigos* é um conjunto de padrões de bits exclusivos que mapeiam para os caracteres de um idioma natural específico. Os produtos da IBM utilizam o termo *página de código* como um sinônimo para conjunto de códigos. Um *território* identifica um código do idioma e determina informações específicas de uma região para o conjunto de códigos especificado. Se estas opções não forem especificadas, o banco de dados utiliza o idioma e a seqüência de intercalação do cliente do DB2 utilizado para criação do banco de dados.

Antes de criar o banco de dados federado, determine quais valores de conjunto de códigos e território serão especificados. Depois de criar o banco de dados, é impossível alterar estes valores. Para escolher um conjunto de códigos para o banco de dados federado, avalie o conjunto de códigos especificado pela origem de dados remota que o banco de dados federado irá acessar. Escolha um conjunto de códigos para o banco de dados federado que corresponda ao conjunto de códigos que as origens de dados remotos utilizam. Se o banco de dados federado for acessar diversas origens de dados, avalie os conjuntos de códigos especificados por todas as origens de dados remotas. Se as origens de dados utilizarem diferentes

conjuntos de códigos ou conjuntos de códigos incompatíveis, especifique Unicode como o conjunto de códigos para o servidor federado.

Para muitas origens de dados, na primeira vez que um wrapper se conecta a uma origem de dados, o wrapper executa estas tarefas:

1. Determina a página de códigos e o território do banco de dados federado.
2. Mapeia o conjunto de códigos e o território a um código do idioma do cliente da origem de dados, se a origem de dados suportar um.
3. Configura uma variável de ambiente, chama uma API da origem de dados para informar à origem de dados qual é o código do idioma do cliente ou prepara para executar a conversão do conjunto de códigos.

A conversão da página de códigos envolve a conversão de dados de caracteres entre a página de códigos do banco de dados da origem de dados e a página de códigos do banco de dados federado. Algumas origens de dados executam a conversão da página de códigos. Para algumas origens de dados que não executam a conversão de página de códigos, o wrapper executa a conversão. Por exemplo, se o banco de dados federado utiliza a página de códigos 819 e o território US, o código do idioma do cliente Oracle equivalente é `American_America.WE8ISO8859P1`. O wrapper do Oracle configura automaticamente a variável de ambiente `NLS_LANG` para o valor do código do idioma do cliente Oracle. Então, quando os dados são enviados do banco de dados Oracle para o wrapper, o banco de dados Oracle converte os dados do conjunto de códigos `American_America.WE8ISO8859P1` para a página de códigos 819. Quando os dados são enviados do wrapper do banco de dados Oracle, o servidor ou o cliente Oracle converte os dados da página de códigos 819 para o conjunto de códigos que o banco de dados Oracle utiliza.

A seqüência de intercalação está relacionada ao idioma que o servidor federado suporta e que o servidor da origem de dados suporta. Para especificar a seqüência de intercalação para o banco de dados federado, inclua a opção `COLLATE USING` no comando `CREATE DATABASE`. Se o banco de dados federado e a origem de dados utilizam a mesma seqüência de intercalação, configure a opção do servidor `COLLATING_SEQUENCE` como 'Y' quando emitir a instrução `CREATE SERVER`. A seqüência de intercalação especificada para o banco de dados federado afeta onde as consultas que envolvem a classificação ou as comparações de caracteres são executadas. Pelo padrão, o banco de dados federado utiliza uma seqüência de intercalação que faz a distinção entre maiúsculas e minúsculas. Porém, algumas origens de dados utilizam uma seqüência de intercalação que não faz distinção entre maiúsculas e minúsculas como o padrão. Uma origem de dados também pode permitir que a seqüência de intercalação seja customizada ou pode oferecer múltiplas opções para a configuração da página de códigos padrão. Se a seqüência de intercalação do banco de dados federado e a origem de dados forem diferentes, uma consulta pode não retornar os resultados esperados. Por exemplo, se a consulta envolver classificação de caracteres, os resultados corretos serão retornados, mas eles não estarão na ordem esperada. Se a consulta envolver comparações de caracteres, resultados incorretos podem ser retornados.

Onde forem executadas operações baseadas em caracteres, haverá um efeito no desempenho da consulta. Quando as seqüências de intercalação forem diferentes, o servidor federado executará a classificação de caracteres e as comparações de caracteres localmente para garantir que sejam retornados conjuntos de resultados classificados consistentemente. Para obter melhores resultados, configure a seqüência de intercalação do banco de dados federado para a mesma seqüência que a origem de dados utiliza. Então, onde possível, o otimizador de consultas

transfere as operações baseadas em caracteres para a origem de dados, para que a origem de dados, não o banco de dados federado, execute as operações.

Configurando o Unicode para Sistemas Federados

Você pode executar wrappers relacionais e não relacionais e funções definidas pelo usuário em um banco de dados na página de códigos Unicode (UTF-8). Um banco de dados na página de códigos Unicode fornece um ambiente de servidor federado que é independente de plataforma.

Suporte Unicode para Sistemas Federados

Todos os wrappers relacionais e não-relacionais e as funções definidas pelo usuário podem ser executadas em um banco de dados na página de códigos Unicode (UTF-8).

O banco de dados na página de código Unicode fornece ambientes de servidor federado que são independentes da plataforma. O banco de dados pode manipular dados que são armazenados em diversas páginas de códigos em diferentes origens de dados.

Na Figura 4 na página 20 uma empresa possui filiais em diferentes países. Cada filial armazena dados dos clientes com seus próprios bancos de dados em sua própria página de códigos. O banco de dados Microsoft SQL Server armazena dados na página de códigos A. O banco de dados Oracle armazena dados na página de códigos B. A página de códigos A e a página de códigos B estão em territórios diferentes. Para integrar os dados de diferentes territórios, a empresa pode configurar a página de códigos do banco de dados federado como Unicode. A empresa pode então unir as tabelas para visualizar o número total de ordens de compra, independente do território.

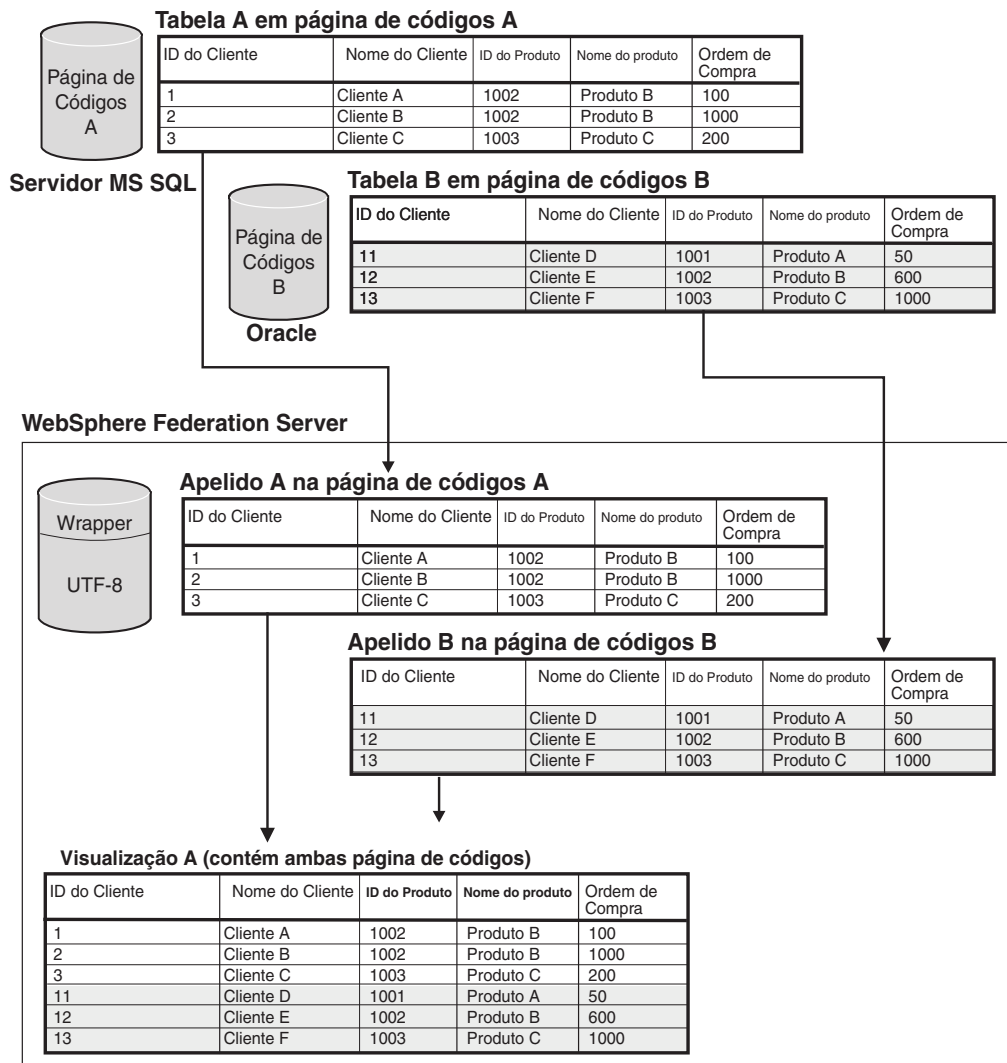


Figura 4. Exemplo Unicode

Especificando a página de código do cliente para suporte Unicode de origens de dados do Microsoft SQL Server e ODBC

Para assegurar uma conversão correta de página de códigos para origens de dados do Microsoft SQL Server e ODBC, é necessário a página de código do cliente se a página de códigos for diferente da página de códigos do banco de dados federado.

Procedimento

Para especificar a página de código do cliente, emita uma instrução CREATE SERVER com a opção CODEPAGE definida para o valor da página de códigos do cliente.

A página de códigos do cliente é a página de códigos de origem de dados. O valor da página de códigos do cliente é o número da página de códigos.

Exemplo: Se a origem de dados for Microsoft SQL Server e o servidor federado estiver no Windows e o código de idioma do sistema padrão nesse sistema operacional estiver configurado para japonês (Shift-JIS), a opção do servidor CODEPAGE deve ser configurada para 943 (Shift-JIS) ou 1202 (UTF-16LE). Para

especificar a página de códigos 1202 para o nome da origem de dados do servidor Microsoft SQL como FEDSERVERW, emita a seguinte instrução:

```
CREATE SERVER FEDSERVERW TYPE MSSQLSERVER VERSION 2000 WRAPPER MSSQLODBC3
  OPTIONS(NODE 'SAMPLE', DBNAME 'TESTDB', CODEPAGE '1202');
```

Exemplo: Se a origem de dados for Microsoft SQL Server e o servidor federado estiver sendo executado no UNIX e a configuração de IANAAppCodePage do cliente DataDirect Connect for 6 (Shift-JIS), a opção do servidor CODEPAGE deve estar configurada para 943 (Shift-JIS) ou 1208 (UTF-8). Para especificar a página de códigos 1208 para a origem de dados do Microsoft SQL server com o nome FEDSERVERU, emita a seguinte instrução:

```
CREATE SERVER FEDSERVERU TYPE MSSQLSERVER VERSION 2000 WRAPPER MSSQLODBC3
  OPTIONS(NODE 'SAMPLE', DBNAME 'TESTDB', CODEPAGE '1208');
```

Páginas de Códigos Unicode Suportadas para a Opção CODEPAGE do Wrapper MSSQL e ODBC

Valores de página de códigos válidos são aqueles que o DB2 Database para Linux, UNIX, e Windows suporta além daqueles mostrados na seguinte tabela.

Tabela 6. Páginas de Códigos Unicode Suportadas para a Opção CODEPAGE do Wrapper MSSQL e ODBC

Valor da opção CODEPAGE	Descrição
1200	Codepage1200 - UCS-2 (big-endian)
1202	Codepage1202 - UCS-2 (little-endian)
1208	Codepage1208 - UTF-8
1232	Codepage1232 - UTF-32 (big-endian)
1234	Codepage1234 - UTF-32 (little-endian)

Especificando a Página de Código do Arquivo para Suporte Unicode de Origens de Dados de Arquivos Estruturados por Tabela

Para assegurar correta conversão de página de códigos para origens de dados do arquivo estruturado por tabela, é necessário especificar a página de códigos do arquivo se a página de códigos for diferente da página de códigos do banco de dados federado.

Sobre Esta Tarefa

Valores válidos são aqueles que o DB2 Database para Linux, UNIX, e Windows suporta. O valor padrão é a página de códigos do banco de dados federado.

Restrições

É possível utilizar a opção CODEPAGE somente em um banco de dados federado Unicode.

Procedimento

Para especificar a página de códigos de um arquivo estruturado por tabela, emita a instrução CREATE NICKNAME com a opção CODEPAGE configurada para o número da página de códigos dos dados no arquivo estruturado por tabela.

Exemplo: A página de códigos dos dados em um arquivo denominado DRUGDATA1.TXT é 943. Para especificar a página de códigos de um arquivo estruturado por tabela como 943, emita a seguinte instrução CREATE NICKNAME:

```
CREATE NICKNAME DRUGDATA1(Dcode Integer NOT NULL, Drug CHAR(20),
    Manufacutuer CHAR(20))
FOR SERVER biochem_lab
OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT',CODEPAGE '943',
COLUMN_DELIMITER '.',
SORTED 'Y', KEY_COLUMN 'DCODE', VALIDATE_DATA_FILE 'Y');
```

Especificando a Página de Códigos do Arquivo para Suporte Unicode das Origens de Dados do Arquivo Estruturado por Tabela - Exemplo

O exemplo a seguir mostra como especificar a página de códigos do arquivo estruturado por tabela.

A página de códigos dos dados em um arquivo denominado DRUGDATA1.TXT é 943. Para especificar a página de códigos de um arquivo estruturado por tabela como 943, emita a seguinte instrução CREATE NICKNAME:

```
CREATE NICKNAME DRUGDATA1(Dcode Integer NOT NULL, Drug CHAR(20),
    Manufacutuer CHAR(20))
FOR SERVER biochem_lab
OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT',CODEPAGE '943',
COLUMN_DELIMITER '.',
SORTED 'Y', KEY_COLUMN 'DCODE', VALIDATE_DATA_FILE 'Y');
```

Erros Quando os Tamanhos do Ponto de Código Remoto e Federado São Diferentes

Quando o tamanho do ponto de código é diferente entre a origem de dados federados e remotos, você pode ter dados truncados retornados ou falhas de inserção ou de atualização.

Quando você seleciona dados da origem de dados remota, esses dados são truncados se a conversão da cadeia de caracteres resultar em um número maior de bytes do que o tamanho da coluna de apelido. Se os dados truncados terminarem em um caractere pendente, os bytes restantes são preenchidos por espaços em branco. Além disso, é possível inserir ou atualizar dados maiores do que o tamanho da coluna de apelido se o tamanho dos dados convertidos for menor do que ou igual ao tamanho da coluna remota.

Se o banco de dados federado tiver um tamanho do ponto de código menor do que a origem de dados remota, a inserção ou atualização dos dados poderá falhar. Inserções ou atualizações falham se a conversão da cadeia de caracteres resultar em um número menor de bytes do que o tamanho da coluna de origem de dados remota.

Ajuste o tamanho da coluna de apelido ou tamanho da coluna de tabela remota para evitar o truncamento de dados ou um erro no truncamento.

Configurando o Servidor Federado para Acessar Origens de Dados

Depois de instalar e configurar o servidor federado e criar um banco de dados federado, planeje e, em seguida, configure o acesso às origens de dados.

O processo para configuração do acesso às origens de dados é o mesmo, independente da origem de dados. O que difere no processo são as configurações particulares aplicadas ao concluir cada tarefa de configuração para cada origem de

dados. O processo descrito aqui é genérico. Para obter informações completas sobre a configuração de uma origem de dados em particular, consulte as informações de configuração detalhadas para a origem de dados.

Reveja como Nomear e Especificar Objetos

Durante o processo de configuração, são utilizadas instruções SQL do DB2 para registrar objetos. Antes de executar qualquer tarefa de configuração, certifique-se de entender as regras de nomenclatura associadas e como a utilização das aspas em instruções SQL afetam a distinção entre maiúsculas e minúsculas dos objetos especificados.

Registre o Wrapper

Depois de utilizar o programa de instalação de federação para instalar o suporte a uma origem de dados, registre o wrapper correspondente. Um wrapper é um conjunto de arquivos de biblioteca que o servidor federado utiliza para se comunicar com a origem de dados e recuperar dados desta origem. Para cada tipo de origem de dados que deseja acessar, registre um wrapper. Por exemplo, para acessar uma tabela em DB2 para Linux, UNIX e Windows, uma tabela em DB2 para iSeries e uma tabela em Teradata, registre dois wrappers: o wrapperDRDA para as origens de dados do DB2 e o wrapper Teradata para as origens de dados Teradata.

Como parte de seu planejamento, decida se o nome padrão do wrapper deve ser utilizado ou não, ou se deve ser designado um nome diferente ao wrapper, e reveja as opções do wrapper que estão disponíveis para cada origem de dados configurada. Cada origem de dados possui uma ou mais opções de wrapper necessárias que devem ser configuradas.

Registre as Definições do Servidor

Antes de poder acessar os objetos de origem de dados específicos, registre uma ou mais definições do servidor. Para uma origem de dados relacional, uma definição de servidor representa um banco de dados remoto, uma partição de banco de dados ou um nó. Para uma origem de dados não-relacional, uma definição de servidor freqüentemente mapeia para outros tipos de objetos de dados externos. Cada origem de dados tem parâmetros exigidos e opcionais que devem ser especificados no registro da definição do servidor.

Como parte de seu planejamento, reveja as opções do servidor que estão disponíveis para cada origem de dados específica configurada. Cada origem de dados possui uma ou mais opções de servidor necessárias que devem ser configuradas.

Registre os Mapeamentos de Usuários

Se uma origem de dados remota necessitar autenticação de usuário e se o ID de usuário e a senha remotos do usuário forem diferente dos utilizados pelo usuário para conexão com o banco de dados federado, defina um mapeamento de usuário. Um mapeamento de usuário é uma associação entre um ID de autorização do servidor federado e um ID e uma senha do usuário da origem de dados. Pelo padrão, os mapeamentos de usuários são armazenados no catálogo no servidor federado.

Como parte de seu planejamento, decida se deseja armazenar informações de mapeamento de usuário em um repositório externo, como em um servidor LDAP ou em um arquivo. Para utilizar um repositório externo, crie um plug-in que forneça ao servidor federado a interface com o repositório.

Atualizar Estatísticas de Origem de Dados

Para cada origem de dados relacional que planejar acessar, utilize um comando que seja equivalente ao comando RUNSTATS do DB2 para atualizar as estatísticas na origem de dados remota. Então, quando criar apelidos, as informações estatísticas mais atualizadas serão incluídas no catálogo do sistema no banco de dados federado. Posteriormente, quando uma consulta for executada na origem de dados, o otimizador de consultas utilizará estas informações para determinar a forma mais eficiente de executar a consulta.

Depois de criar os apelidos, as estatísticas na origem de dados podem ser alteradas. Quando as estatísticas para uma origem de dados relacional forem alteradas, utilize o procedimento armazenado SYSPROC.NNSTAT para atualizar as informações estatísticas no catálogo do sistema. Quando as estatísticas para uma origem de dados não-relacional forem alteradas, utilize a ferramenta que é fornecida pela origem de dados não-relacional, ou atualize manualmente as estatísticas nas visualizações do catálogo SYSTAT.

Registre os Apelidos

Um apelido é criado para cada objeto da origem de dados relacional que desejar acessar. Para algumas origens de dados não-relacionais, defina uma lista fixa de colunas de entrada e saída quando registrar o apelido. Cada coluna especificada é mapeada para um campo, coluna ou elemento particular no objeto de origem de dados.

Como parte de seu planejamento, reveja as opções do apelido e da coluna que estão disponíveis para cada origem de dados configurada. Algumas origens de dados necessitaram de opções de apelido e coluna que foram configuradas.

Execute as Tarefas de Configuração Adicionais

Dependendo de como deseja trabalhar com a origem de dados, você pode querer executar estas tarefas de configuração adicionais.

Crie especificações de índice

É possível definir uma especificação de índice para objetos que não possuem um índice. Por exemplo, uma especificação de índice é criada quando uma tabela adquire um novo índice ou se o objeto de origem de dados, como uma visualização, não possui um índice.

Defina mapeamentos de tipos de dados alternativos

No sistema federado, existem mapeamentos padrão entre os tipos de dados da origem de dados e os tipos de dados do banco de dados federado. Para origens de dados relacionais, é possível definir mapeamentos de tipos de dados alternativos. Por exemplo, é possível alterar um mapeamento de tipo para todos os objetos de origem de dados localizados em um servidor específico ou alterar um mapeamento de tipo para um objeto de origem de dados específico, tipo de origem de dados ou objeto de origem de dados e tipo.

Defina mapeamentos de funções alternativos

No sistema federado há mapeamentos de funções padrão entre as funções integradas da origem de dados e as funções integradas do banco de dados federado. Para origens de dados relacionais, é possível definir mapeamentos de funções alternativos. Por exemplo, é possível definir um mapeamento de função alternativo quando desejar utilizar uma nova função integrada ou uma função definida pelo usuário que está disponível na origem de dados mas para a qual o banco de dados federado necessita de uma função mapeada.

Distinção entre maiúsculas e minúsculas e uso correto de aspas

Quando valores de opções e objetos são especificados em instruções SQL do DB2, é necessário saber quando as aspas são obrigatórias, que tipo utilizar e como afetam a distinção entre maiúsculas e minúsculas.

A forma como os objetos são nomeados quando são criados afeta a condição dos caracteres maiúsculos e minúsculos no nome do objeto e determina como os nomes de objetos e valores das opções são determinados nos comandos. Por exemplo, se ao criar um apelido o nome não for colocado entre aspas duplas, o catálogo do sistema armazenará o apelido como caracteres maiúsculos, independentemente dos caracteres utilizados para nomear o objeto. Se forem utilizadas aspas duplas na criação do apelido, o catálogo armazenará os caracteres do nome do objeto exatamente da forma especificada. Assim, ao utilizar o nome do objeto como um valor de opção, especifique exatamente os caracteres maiúsculos e minúsculos. Por exemplo, a opção de coluna FOREIGN_KEY que o Script, os Serviços da Web e os Wrappers XML suportam exigem que o apelido seja especificado para a coluna chave estrangeira como o valor da opção. Quando o valor da opção for inserido, utilize a mesma combinação de caracteres maiúsculos e minúsculos que o catálogo do servidor federado utiliza para armazenar o apelido.

A tabela a seguir descreve a utilização correta de caracteres maiúsculos e minúsculos e aspas quando os valores de opções e objetos são especificados em instruções SQL do DB2.

Tabela 7. Utilização correta de maiúsculas e minúsculas e aspas

Identificador	Utilização de maiúsculas e minúsculas e aspas	Exemplos
Valor da opção	Utilize a combinação de maiúsculas e minúsculas que o valor da opção exige, e coloque o valor da opção entre aspas simples.	<p>Esta instrução cria uma tabela de origens de dados nomeada como <code>remote_schema.remote_table</code> (tudo em minúsculas):</p> <pre>CREATE TABLE newton.my_nick (c1 int) OPTIONS (remote_server 'MY_SERVER' remote_schema 'remote_schema', remote_tabname 'remote_table');</pre> <p>Esta instrução cria uma tabela de origens de dados nomeada como <code>REMOTE_SCHEMA.REMOTE_TABLE</code> (tudo em maiúsculas):</p> <pre>CREATE TABLE newton.my_nick (c1 int) OPTIONS (remote_server'MY_SERVER' remote_schema 'REMOTE_SCHEMA', remote_tabname 'REMOTE_TABLE');</pre>
Objeto que contém somente caracteres minúsculos	Utilize todos os caracteres minúsculos e coloque o identificador entre aspas duplas.	<p>Esta instrução cria um apelido em uma tabela de origem de dados nomeada como <code>infx_user.remote_table</code> (tudo em minúsculas):</p> <pre>CREATE NICKNAME my_nick FOR infx_server. "infx_user"."remote_table";</pre> <p>Nota: Algumas origens de dados como Informix e Teradata utilizam nomes em caracteres minúsculos por padrão.</p>
O objeto que contém somente caracteres maiúsculos, números e caracteres sublinhado (<code>_</code>)	<p>Há duas opções:</p> <ul style="list-style-type: none"> • Utilize todos os caracteres maiúsculos e coloque o identificador entre aspas duplas. • Utilize qualquer combinação de caracteres maiúsculos e minúsculos e coloque o identificador entre aspas duplas. 	<p>Cada uma destas instruções cria o apelido <code>MY_NICK</code> (tudo em maiúsculas):</p> <pre>CREATE NICKNAME my_nick FOR infx_server. "infx_user"."remote_table";</pre> <pre>CREATE NICKNAME "MY_NICK" FOR infx_server. "infx_user"."remote_table";</pre>

Para IDs e senhas de autorização de origens de dados, também é possível utilizar as opções de servidor `FOLD_ID` e `FOLD_PW` para converter o ID e a senha para a combinação correta de maiúsculas e minúsculas.

A partir de um prompt de comandos do sistema operacional UNIX

Se um valor que faz distinção entre maiúsculas e minúsculas for colocado entre aspas no prompt de comandos do UNIX no servidor federado, certifique-se de que as aspas estejam sendo analisadas corretamente:

Instruções SQL que contêm aspas duplas, mas que não contêm aspas simples

Se a instrução SQL contiver aspas duplas mas não contiver aspas simples, coloque toda a instrução entre aspas simples.

Por exemplo, se desejar emitir esta instrução SQL:

```
CREATE NICKNAME my_nickname FOR my_server."proprietário"."my_table"
```

Digite o texto a seguir no prompt de comandos doUNIX:

```
db2 'CREATE NICKNAME my_nickname FOR my_server."proprietário"."my_table"'
```

Instruções SQL que contêm aspas simples, mas que não contêm aspas duplas

Se a instrução SQL contiver aspas simples mas não contiver aspas duplas, coloque toda a instrução entre aspas duplas.

Por exemplo, se desejar emitir esta instrução SQL:

```
CREATE USER MAPPING FOR USER SERVER my_server  
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

Digite o texto a seguir no prompt de comandos doUNIX:

```
db2 "CREATE USER MAPPING FOR USER SERVER my_server  
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')"
```

Instruções SQL que contêm aspas duplas e simples

Se a instrução SQL contém aspas simples e duplas:

- Coloque toda a instrução em aspas duplas
- Preceda cada aspa dupla na instrução com uma barra invertida.

Por exemplo, para emitir esta instrução SQL:

```
CREATE USER MAPPING FOR "local_id" SERVER my_server  
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

Digite o texto a seguir no prompt de comandos doUNIX:

```
db2 "CREATE USER MAPPING FOR \"local_id\" SERVER my_server  
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')"
```

Os exemplos acima assumem que as instruções SQL estão sendo digitadas a partir do prompt de comandos UNIX e passadas para o comando doDB2, sem a utilização a opção -f. Para utilizar o comando doDB2 com a opção -f para inserção de instruções SQL a partir de um arquivo, digite as instruções como mostrado na primeira ocorrência de cada exemplo.

A partir de um prompt de comandos do sistema operacional Windows

Para preservar valores que fazem a distinção entre maiúsculas e minúsculas ao digitar comandos a partir de um prompt de comandos Microsoft Windows no servidor federado, preceda cada aspas duplas por uma barra invertida. Por exemplo, para criar o apelido *nick1* para a tabela *weekly_salary* do Microsoft SQL Server. A tabela se encontra no banco de dados *NORBASE*. O esquema local é *my_schema*.

No prompt de comandos doWindows no servidor federado, digite:

```
db2 CREATE NICKNAME nick1  
  FOR NORBASE."my_schema".\"weekly_salary\"
```

A partir da linha de comandos doDB2 ou de um aplicativo

Quando um valor é especificado a partir da linha de comandos doDB2 ou de um aplicativo, é possível preservar os valores que fazem distinção entre maiúsculas e minúsculas colocando-os entre as aspas apropriadas.

Por exemplo, você deseja criar um mapeamento de usuário para o ID de usuário *local_id*. O ID de usuário remoto *my_id* e a senha remota *my_password*. Você deseja que todos esses três valores sejam conservados em letras minúsculas. No prompt de comandos doDB2 digite:

```
CREATE USER MAPPING FOR "local_id" SERVER my_server  
OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

Configurando Vários Servidores Federados para Acessar Origens de dados

Um sistema federado pode consistir em vários servidores federados. Em vez de configurar cada servidor federado separadamente, é possível poupar tempo usando o IBM Data Studio para configurar os servidores federados.

Antes de Iniciar

- A federação deve ser instalada em um servidor que atuará como o servidor federado
- Um banco de dados federado deve existir no servidor federado.

Sobre Esta Tarefa

Para as origens de dados DB2 suportadas e para a origem de dados Oracle, é possível usar o Explorer do Data Studio Administration para configurar os objetos federados para a origem de dados que você deseja acessar. Após configurar o primeiro servidor, será possível reutilizar ou modificar essas instruções para configurar rapidamente os servidores federados adicionais.

Também será possível usar o editor de SQL e XQuery do Data Studio para configurar diversos servidores para qualquer origem de dados.

Procedimento

Configure diversos servidores federados usando um dos métodos a seguir.

- Para as origens de dados DB2 e a origem de dados Oracle suportadas, siga as instruções em Criando e gerenciando diversos objetos federados.
- Para outras origens de dados, siga as instruções em Criando e gerenciando objetos federados: Editor de SQL e XQuery.

Capítulo 2. Configurando Origens de Dados

Você pode configurar o servidor federado para acessar as origens de dados relacional e não relacional.

Configurando o Acesso às Origens de Dados do BioRS

Você pode integrar os dados que estão nos bancos de dados BioRS com informações de outras origens utilizando um sistema federado.

Sobre Esta Tarefa

Para configurar um servidor federado para acessar as origens de dados BioRS, é necessário fornecer ao servidor federado informações sobre as origens de dados e objetos que você deseja acessar. Depois de configurar o servidor federado, você poderá criar consultas e utilizar as funções customizadas para acessar as origens de dados BioRS.

Wrapper BioRS

BioRS é um sistema de consulta e recuperação que pode ser utilizado para recuperar informações de várias origens de dados. O wrapper BioRS utiliza as mesmas APIs que a interface baseada na Web do BioRS para executar consultas.

BioRS é um sistema de consulta e recuperação desenvolvido pela Biomax Informatics. Você pode utilizar o BioRS para recuperar informações de várias origens de dados, incluindo arquivos simples e bancos de dados relacionais. Geralmente, você faz download de dados públicos, tais como, SwissProt e GenBank, como arquivos simples em seu sistema BioRS. O BioRS pode integrar origens de dados públicas e proprietárias (por exemplo, bancos de dados privados que são mantidos por sua organização) em um ambiente comum.

Quando uma origem de dados tiver sido integrada ao sistema BioRS, ela será referida como um *banco de dados*. Os elementos que estão contidos em cada entrada do banco de dados são coletivamente referidos como um *esquema*. Os elementos de um banco de dados indexados podem ser utilizados nas funções BIOR.S.CONTAINS, BIOR.S.CONTAINS_GE e BIOR.S.CONTAINS_LE. As funções BioRS são especificadas na cláusula WHERE da instrução SELECT. Os elementos que não estão indexados podem ser referidos na lista SELECT e em outros predicados na cláusula WHERE. Os elementos que não estão indexados serão processados pelo servidor federado.

Você pode estabelecer relacionamentos entre entradas em bancos de dados para poder vincular bancos de dados no sistema BioRS.

Os bancos de dados BioRS podem ter um relacionamento pai-filho (bancos de dados podem ser aninhados). Neste relacionamento, o banco de dados filho contém um elemento de tipo de dados Reference chamado PARENT. O elemento PARENT refere-se ao elemento `_ID_` do banco de dados pai. Diferente da presença deste elemento PARENT predefinido, os bancos de dados aninhados contêm os mesmos dados que bancos de dados não aninhados.

O BioRS fornece uma interface baseada na Web que permite que os usuários executem consultas nos dados em bancos de dados BioRS. O wrapper BioRS utiliza as mesmas APIs (Interfaces de Programação de Aplicativos) que a interface baseada na Web do BioRS para executar consultas.

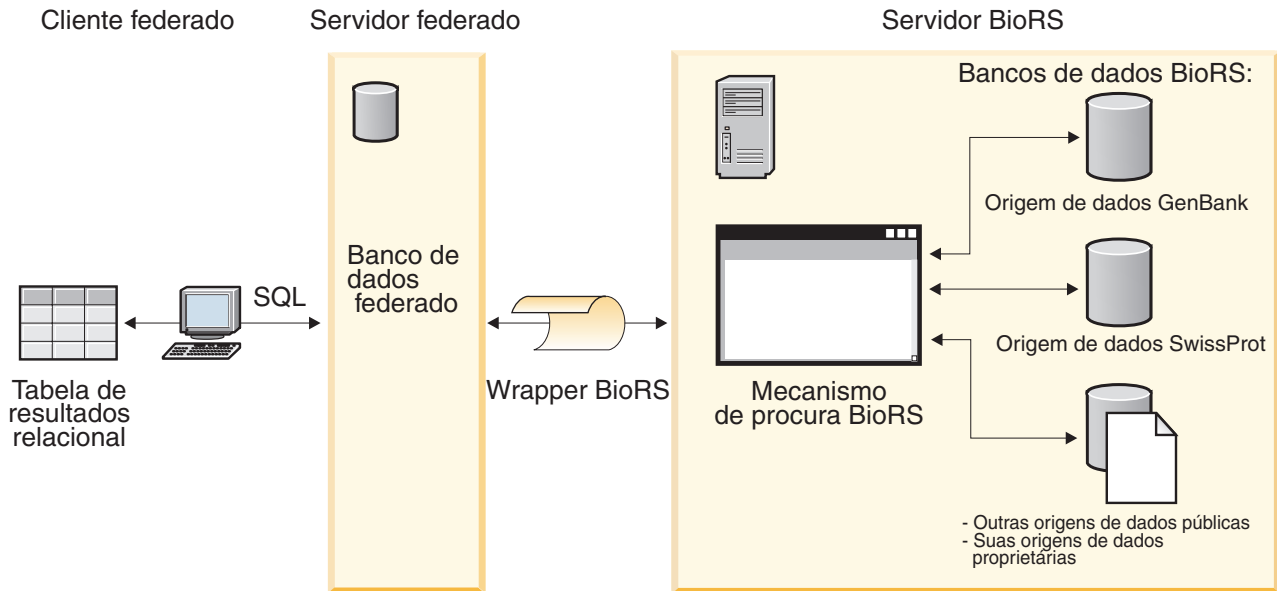


Figura 5. Como Funciona o Wrapper BioRS

A partir do cliente, os usuários ou aplicativos enviam uma consulta utilizando instruções SQL. Em seguida, a consulta é enviada para seu sistema federado no qual o wrapper BioRS está instalado. Dependendo de como a consulta foi construída, o servidor federado e o servidor BioRS podem ser utilizados para processar a consulta. O servidor BioRS pode estar em um computador diferente do sistema federado. As informações sobre autenticação devem ser fornecidas pelo sistema federado para o servidor BioRS para cada consulta. Estas informações podem ser uma combinação de ID do usuário e senha, ou uma indicação não autenticada (geralmente uma conta guest).

Para obter informações detalhadas sobre o produto BioRS, consulte o Web site da Biomax em: <http://www.biomax.com>.

Incluindo Origens de Dados BioRS em um Servidor Federado

Para configurar um servidor federado para acessar origens de dados BioRS, é necessário fornecer ao servidor federado informações sobre as origens de dados e objetos que você deseja acessar.

Antes de Iniciar

- A federação deve ser instalada em um servidor que atuará como o servidor federado.
- Um banco de dados deve existir no servidor federado.

Procedimento

1. Registre as funções customizadas para o wrapper BioRS.
2. Registre o wrapper BioRS.
3. Registre as definições do servidor BioRS.

4. Opcional: Crie os mapeamentos de usuário BioRS.
5. Registre apelidos para bancos de dados BioRS.

Registrando as Funções Customizadas para o Wrapper BioRS

Você deve registrar as funções customizadas BioRS antes de registrar o wrapper BioRS. As funções customizadas BioRS são utilizadas com o wrapper BioRS para transferir predicados para o mecanismo de consulta BioRS.

Sobre Esta Tarefa

Você deve registrar todas as funções customizadas em cada instância de banco de dados federado na qual o wrapper BioRS está instalado.

Todas as funções customizadas para o wrapper BioRS devem ser registradas com o nome do esquema `biors`.

Procedimento

Para cada uma das funções customizadas BioRS, emita a instrução `CREATE FUNCTION`. Você deve especificar dois tipos de dados em cada função customizada. O primeiro tipo de dados especificado é a coluna indexada. O segundo tipo de dados especificado é o termo de procura. Você deve incluir as palavras-chave `AS TEMPLATE`, `DETERMINISTIC` e `NO EXTERNAL ACTION` na instrução `CREATE FUNCTION`.

O exemplo a seguir mostra a sintaxe para a função `BIORS.CONTAINS`:

```
CREATE FUNCTION biors.contains (column_data_type, search_term_data_type)
  RETURNS INTEGER AS TEMPLATE
  DETERMINISTIC NO EXTERNAL ACTION;
```

O que Fazer Depois

Dica: Utilize o arquivo de amostra `create_function_mappings.ddl` para registrar as funções customizadas BioRS. O arquivo de amostra está localizado no diretório `sqllib/samples/lifesci/biors` no servidor federado. O arquivo de amostra contém as instruções `CREATE FUNCTION` para cada uma das possíveis combinações de tipos de dados. Para registrar as funções customizadas, edite o arquivo `create_function_mappings.ddl` para especificar os tipos de dados para as colunas de índice e termos de procura para cada função customizada. Em seguida, você deve executar o arquivo `create_function_mappings.ddl` em cada instância de banco de dados federado na qual o wrapper BioRS está instalado.

Funções Customizadas para o Wrapper BioRS:

Descrições e exemplos das funções customizadas que são utilizadas com o wrapper BioRS.

O ambiente federado utiliza mecanismos de consulta. Para o wrapper BioRS, estes mecanismos de consulta são o mecanismo de consulta do banco de dados federado e o mecanismo de consulta BioRS. Você pode especificar que predicados sejam transferidos para o mecanismo do BioRS utilizando as funções customizadas BioRS.

As funções customizadas para o wrapper BioRS são:

- `BIORS.CONTAINS`
- `BIORS.CONTAINS_LE`

- BIRS.CONTAINS_GE
- BIRS.SEARCH_TERM

Você utiliza a instrução CREATE FUNCTION para registrar as funções customizadas BioRS.

A tabela a seguir lista as quatro funções customizadas BioRS com exemplos de tipos de dados válidos que podem ser especificados ao registrar as funções.

Tabela 8. Funções Customizadas para o Wrapper BioRS

Função	Descrição
BIORS.CONTAINS (VARCHAR(), VARCHAR()) BIORS.CONTAINS (VARCHAR(), CHAR()) BIORS.CONTAINS (VARCHAR(), DATE) BIORS.CONTAINS (VARCHAR(), TIMESTAMP)	Procura em uma coluna indexada valores que sejam iguais (de acordo com a semântica da consulta BioRS) ao valor especificado. O primeiro argumento deve ser uma referência à coluna indexada e o segundo argumento é o valor de procura especificado.
BIORS.CONTAINS_LE (VARCHAR(), VARCHAR()) BIORS.CONTAINS_LE (VARCHAR(), SMALLINT) BIORS.CONTAINS_LE (VARCHAR(), BIGINT) BIORS.CONTAINS_LE (VARCHAR(), DECIMAL) BIORS.CONTAINS_LE (VARCHAR(), DOUBLE) BIORS.CONTAINS_LE (VARCHAR(), REAL)	Procura em uma coluna indexada valores que sejam menores ou iguais (de acordo com a semântica da consulta BioRS) ao valor especificado. O primeiro argumento deve ser uma referência à coluna indexada e o segundo argumento é o valor de procura especificado.
BIORS.CONTAINS_GE (CHAR(), CHAR()) BIORS.CONTAINS_GE (CHAR(), DATE) BIORS.CONTAINS_GE (CHAR(), TIMESTAMP) BIORS.CONTAINS_GE (CHAR(), INTEGER) BIORS.CONTAINS_GE (CHAR(), SMALLINT) BIORS.CONTAINS_GE (CLOB(), DATE)	Procura em uma coluna indexada valores que sejam maiores ou iguais (de acordo com a semântica da consulta BioRS) ao valor especificado. O primeiro argumento deve ser uma referência à coluna indexada e o segundo argumento é o valor de procura especificado.
BIORS.SEARCH_TERM (VARCHAR(), VARCHAR()) BIORS.SEARCH_TERM (VARCHAR(), CHAR()) BIORS.SEARCH_TERM (CHAR(), VARCHAR()) BIORS.SEARCH_TERM (CHAR(), CHAR())	Transmite uma expressão de consulta BioRS para o mecanismo de procura do BioRS.

Registrando o Wrapper BioRS

Você deve registrar um wrapper para acessar origens de dados BioRS. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Sobre Esta Tarefa

Se você utilizar um servidor proxy para acessar arquivos BioRS, deverá especificar as informações do proxy como opções quando registrar o wrapper. Por padrão, as opções do wrapper são utilizadas quando você consulta qualquer arquivo BioRS. No entanto, se você também especificar as opções de proxy ao registrar uma definição de wrapper, as opções do servidor têm prioridade.

Procedimento

Emita a instrução CRIAR WRAPPER a partir da linha de comandos.
Por Exemplo:

```
CREATE WRAPPER wrapper_name
LIBRARY library_name;
```


Se você utilizar um servidor proxy para acessar documentos do BioRS, a instrução emitida será:

```
CREATE WRAPPER wrapper_name  
LIBRARY library_name  
OPTIONS (PROXY_TYPE 'type',  
PROXY_SERVER_NAME 'server_name',  
PROXY_SERVER_PORT 'port_number');
```

Especifique o parâmetro LIBRARY na instrução CREATE WRAPPER. O nome do arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado. Consulte a lista de arquivos da biblioteca do wrapper BioRS para obter o nome correto da biblioteca a ser especificado na instrução CREATE WRAPPER.

Arquivos da Biblioteca do Wrapper BioRS:

Os arquivos da biblioteca do wrapper BioRS são incluídos no servidor federado quando você instala o servidor federado.

Quando você instala o servidor federado, três arquivos de biblioteca são incluídos no caminho do diretório padrão. Por exemplo, se o servidor federado estiver em execução no AIX, os arquivos da biblioteca do wrapper incluídos no caminho do diretório serão libdb2lsbiors.a, libdb2lsbiorsF.a e libdb2lsbiorsU.a. O arquivo da biblioteca do wrapper padrão é libdb2lsbiors.a. Os outros arquivos da biblioteca do wrapper são utilizados com opções de wrapper específicas.

Inclua o parâmetro LIBRARY na instrução CREATE WRAPPER e especifique o nome do arquivo da biblioteca padrão do wrapper.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 9. Locais e Nomes de Arquivos da Biblioteca de Wrapper BioRS

Sistema Operacional	Caminho do Diretório	Nome do arquivo da biblioteca do wrapper
AIX	/usr/opt/<install_path>/lib32/ /usr/opt/<install_path>/lib64/	libdb2lsbiors.a
Linux	/opt/IBM/db2/<install_path>/lib32 /opt/IBM/db2/<install_path>/lib64	libdb2lsbiors.so
Solaris	/opt/IBM/db2/<install_path>/lib32 /opt/IBM/db2/<install_path>/lib64	libdb2lsbiors.so
Windows	%DB2PATH%\bin	db2lsbiors.dll

<install_path> é o caminho do diretório onde o servidor federado está instalado no UNIX ou no Linux.

%DB2PATH% é a variável de ambiente utilizada para especificar o caminho do diretório em que o servidor federado está instalado no Windows. O caminho padrão do diretório do Windows é C:\Arquivos de Programa\IBM\SQLLIB.

Instrução CREATE WRAPPER - Exemplos para o Wrapper BioRS:

Utilize a instrução CREATE WRAPPER para registrar o wrapper BioRS. Os exemplos mostram as opções necessárias para acessar documentos BioRS com e sem um servidor proxy.

Registrando um Wrapper

Se não estiver utilizando um servidor proxy para acessar documentos do BioRS, emita esta instrução para registrar o wrapper:

```
CREATE WRAPPER biors_wrapper LIBRARY 'libdb2lsbiors.a';
```

biors_wrapper

O nome designado a wrapper BioRS. Nomes de wrapper duplicados não são permitidos.

LIBRARY '*libdb2lsbiors.a*'

O nome do arquivo de biblioteca para servidores federados que usam sistemas operacionais AIX.

Registrando um Wrapper para um Servidor Proxy HTTP

Para registrar um wrapper e especificar um servidor proxy HTTP, utilize a seguinte instrução:

```
CREATE WRAPPER biors_proxy LIBRARY 'libdb2lsbiors.a'  
  OPTIONS (PROXY_TYPE 'HTTP',  
          PROXY_SERVER_NAME 'proxy.mysite.com',  
          PROXY_SERVER_PORT '81');
```

PROXY_TYPE '*HTTP*'

Especifica o tipo de proxy que é utilizado para acessar a Internet quando o servidor federado está por trás de um firewall. Os valores válidos são 'NONE', 'HTTP' ou 'SOCKS'.

PROXY_SERVER_NAME '*proxy.mysite.com*'

Especifica o nome do servidor proxy ou o endereço IP. Esta opção será necessária se o valor para a opção do servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

PROXY_SERVER_PORT '*81*'

Especifica o número da porta do servidor proxy. Esta opção será necessária se o valor para a opção do servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

Registrando um Wrapper para um Servidor Proxy SOCKS

Para registrar um wrapper e especificar um servidor proxy SOCKS sem informações sobre autenticação, use a seguinte instrução:

```
CREATE WRAPPER biors_wrapper LIBRARY 'libdb2lsbiors.so'  
  OPTIONS (PROXY_TYPE 'SOCKS',  
          PROXY_SERVER_NAME 'proxy_socks',  
          PROXY_SERVER_PORT '1081');
```

LIBRARY '*libdb2lsbiors.so*'

O nome do arquivo da biblioteca do wrapper para servidores federados que utilizam os sistemas operacionais Linux e Solaris.

PROXY_TYPE '*SOCKS*'

Especifica o tipo de proxy que é utilizado para acessar a Internet quando o servidor federado está por trás de um firewall. Os valores válidos são 'NONE', 'HTTP' ou 'SOCKS'.

PROXY_SERVER_NAME '*proxy_socks*'

Especifica o nome do servidor proxy ou o endereço IP. Esta opção será necessária se o valor para a opção do servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

`PROXY_SERVER_PORT '1081'`

Especifica o número da porta do servidor proxy. Esta opção será necessária se o valor para a opção do servidor `PROXY_TYPE` for 'HTTP' ou 'SOCKS'.

Registrando a Definição do Servidor para uma Origem de Dados BioRS

Você deve registrar cada servidor BioRS que deseja acessar no banco de dados federado.

Sobre Esta Tarefa

Se você utilizar um servidor proxy para acessar arquivos BioRS, poderá especificar as informações do proxy como opções quando registrar a definição do servidor. Se você especificar as informações do proxy na definição do servidor, as opções do servidor terão precedência sobre as opções especificadas ao registrar o wrapper.

Procedimento

Emita a instrução `CRIAR SERVIDOR` a partir da linha de comandos.

Por Exemplo:

```
CREATE SERVER server_definition_name
VERSION version_number
WRAPPER wrapper_name
OPTIONS (NODE 'node_name');
```

Se você utilizar um servidor proxy para acessar arquivos BioRS, deverá especificar várias opções quando registrar o wrapper BioRS ou a definição do servidor. Para especificar as informações do servidor proxy quando registrar o wrapper BioRS, emita esta instrução:

```
CREATE SERVER server_definition_name
VERSION version_number
WRAPPER wrapper_name
OPTIONS (PROXY_TYPE 'type',
PROXY_SERVER_NAME 'server_name',
PROXY_SERVER_PORT 'port_number');
```

Quando registrar a definição do servidor, especifique opções do servidor na instrução `CREATE SERVER`. Existem opções do servidor necessárias e opcionais. A opção do servidor `NODE` é necessária.

O que Fazer Depois

Após o registro da definição do servidor, utilize a instrução `ALTER SERVER` para incluir ou descartar opções do servidor.

Instrução `CREATE SERVER` - Exemplos para o Wrapper BioRS:

Utilize a instrução `CREATE SERVER` para registrar definições do servidor para o wrapper BioRS. Os exemplos mostram os parâmetros necessários, os parâmetros opcionais e as opções do servidor adicionais que podem ser especificadas.

Registrando uma Definição do Servidor com os Parâmetros Necessários

O exemplo a seguir mostra como registrar uma definição do servidor para um wrapper BioRS emitindo a instrução `CREATE SERVER`:

```
CREATE SERVER biors_server VERSION '5.2' WRAPPER biors_wrapper
OPTIONS(NODE 'biors_myco.com');
```

biors_server

Um nome designado ao servidor BioRS. Nomes de definição de servidor duplicados não são permitidos.

VERSION '5.2'

A versão do servidor BioRS que você deseja acessar. As versões suportadas do BioRS são 5.0.14 e 5.2. Se estiver acessando um servidor BioRS que esteja na versão 5.2, será necessário especificar '5.2' como o valor para o parâmetro VERSION. Não será necessário especificar esta opção se você estiver utilizando a versão 5.0.14. O valor padrão de '1.0', que equivale à versão 5.0.14, será utilizado para este parâmetro se você não especificar o valor.

WRAPPER *biors_wrapper*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

NODE '*biors_myco.com*'

Especifica o nome do host do sistema no qual a ferramenta de consulta BioRS está disponível. O valor padrão é *localhost*. Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do nó esteja especificado como uma opção na instrução CREATE SERVER, ele é necessário para origens de dados BioRS.

Registrando uma Definição do Servidor Utilizando Parâmetros Opcionais e Opções do Servidor

O exemplo a seguir mostra parâmetros adicionais e opções do servidor que você pode especificar quando registrar uma definição do servidor para um wrapper BioRS:

```
CREATE SERVER biors_server TYPE BioRS VERSION '5.2'  
  WRAPPER biors_wrapper  
  OPTIONS (NODE 'biors_server2.com', PORT '5555', TIMEOUT 30 ,  
  CASE_SENSITIVE 'N');
```

TYPE *BioRS*

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso. Para o wrapper BioRS, o tipo de servidor deve ser BioRS. Este parâmetro é opcional.

PORT '5555'

Especifica o número da porta a ser utilizada para conectar-se ao servidor BioRS. O valor padrão é '5014'.

TIMEOUT 30

Especifica o tempo, em minutos, que o wrapper BioRS aguarda uma resposta do servidor BioRS. O valor padrão é 10. Este parâmetro é opcional.

CASE_SENSITIVE 'N'

Especifica se o servidor BioRS faz distinção entre maiúsculas e minúsculas em relação aos nomes. Os valores válidos são 'Y' ou 'N'. O valor padrão é 'Y'.

No produto BioRS, um parâmetro de configuração controla a distinção entre maiúsculas e minúsculas dos dados que estão armazenados na máquina servidor BioRS. A opção CASE_SENSITIVE é a duplicata do servidor federado para este parâmetro de configuração do sistema BioRS. Você deve sincronizar as definições de configuração de distinção entre maiúsculas e minúsculas do servidor BioRS em seu sistema BioRS e no servidor federado. Se você não mantiver as definições de configuração de

distinção entre maiúsculas e minúsculas sincronizadas entre o servidor BioRS e o servidor federado, ocorrerão erros ao tentar acessar os dados do BioRS por meio do servidor federado.

Importante: Não é possível alterar ou excluir a opção `CASE_SENSITIVE` após a criação de uma nova definição BioRS. Se precisar alterar a opção `CASE_SENSITIVE`, será necessário eliminar e, em seguida, criar novamente a definição do servidor. Se você eliminar a definição do servidor BioRS, também deverá criar todos os apelidos BioRS que fazem referência a essa definição do servidor. O servidor federado elimina automaticamente todos os apelidos que correspondem a um servidor eliminado.

Registrando uma Definição do Servidor que Inclui um Servidor Proxy

```
CREATE SERVER biors_proxy_serv VERSION 5.2 WRAPPER biors_proxy
  OPTIONS (NODE 'biors.mysite.com',
    PORT '5555',
    PROXY_TYPE 'HTTP'
    PROXY_SERVER_NAME 'proxy.mysite.com
    PROXY_SERVER_PORT '81'
```

PROXY_TYPE 'HTTP'

Especifica o tipo de proxy que é utilizado para acessar a Internet quando se está por trás de um firewall. Os valores válidos são 'NONE', 'HTTP' ou 'SOCKS'.

PROXY_SERVER_NAME 'proxy.mysite.com'

Especifica o nome do servidor proxy ou o endereço IP. Esta opção será necessária se o valor para a opção do servidor `PROXY_TYPE` for 'HTTP' ou 'SOCKS'.

PROXY_SERVER_PORT '81'

Especifica o número da porta do servidor proxy. Esta opção será necessária se o valor para a opção do servidor `PROXY_TYPE` for 'HTTP' ou 'SOCKS'.

Registrando uma Definição do Servidor que Inclui um Servidor Proxy com Informações sobre Autenticação

Para registrar uma definição do servidor e especificar um servidor proxy SOCKS com informações sobre autenticação, utilize a seguinte instrução:

```
CREATE SERVER biors_proxy_serv VERSION 5.2 WRAPPER biors_proxy
  OPTIONS (NODE 'biors.mysite.com',
    PORT '5555',
    PROXY_TYPE 'SOCKS'
    PROXY_SERVER_NAME 'proxy_socks',
    PROXY_SERVER_PORT '1081',
    PROXY_AUTHID 'argle'
    PROXY_PASSWORD 'bargle')
```

PROXY_AUTHID 'argle'

Especifica o ID do usuário no servidor proxy. Esta opção do servidor será necessária se o valor da opção de servidor `PROXY_TYPE` for 'SOCKS'.

PROXY_PASSWORD 'bargle'

Especifica a senha no servidor proxy que está associada ao nome de usuário '*argle*'. Esta opção do servidor será necessária se o valor da opção de servidor `PROXY_TYPE` for 'SOCKS'.

Criando os Mapeamentos de Usuário para uma Origem de Dados BioRS

Quando tentar acessar um servidor BioRS, o servidor federado estabelecerá uma conexão com o servidor BioRS. Dependendo dos métodos de acesso da conta utilizados em seu sistema BioRS, pode ser necessário criar mapeamentos de usuário.

Antes de Iniciar

Um mapeamento de usuário é uma associação entre o ID e a senha do usuário de cada servidor federado e o ID e a senha do usuário da origem de dados correspondente.

Existem dois métodos para especificar mapeamentos do usuário com sistemas federados. Você pode utilizar um repositório externo, como o LDAP, para armazenar os mapeamentos do usuários ou criar os mapeamentos do usuário no catálogo do banco de dados federado.

Os métodos de acesso da conta utilizados em seu sistema BioRS determinarão se é necessário criar mapeamentos de usuário:

- Se seu servidor BioRS estiver configurado para acesso guest para todas as contas de usuário, não será necessário criar mapeamentos de usuário. O servidor federado utiliza uma conta guest para acessar o servidor BioRS.
- Se você tiver um repositório externo, como o LDAP, para armazenar mapeamentos de usuário, não será necessário criar mapeamentos de usuário. É necessário especificar a opção DB2_UM_PLUGIN no wrapper BioRS. Você pode especificar essa opção quando registrar ou alterar o wrapper. O esquema no repositório externo deve incluir acesso guest.
- Se seu servidor BioRS estiver configurado para autenticar contas de usuário com IDs e senhas, será necessário criar mapeamentos de usuário no banco de dados federado para contas de usuário que utilizarão o wrapper BioRS para acessar origens de dados BioRS.
- Se seu servidor BioRS estiver configurado para utilizar uma combinação de contas de usuário guest e autenticado, será necessário criar mapeamentos de usuário para as contas de usuário autenticado que utilizarão o wrapper BioRS para acessar origens de dados BioRS.

Sobre Esta Tarefa

Os mapeamentos de usuário fornecem uma maneira de autenticar o acesso de usuários ou aplicativos que consultam uma origem de dados BioRS com o wrapper BioRS. Se um mapeamento de usuário não estiver definido para um usuário ou aplicativo, o servidor federado utilizará uma conta guest. Se um banco de dados que está sendo consultado precisar de autenticação, poderá ser retornada uma mensagem de erro.

Para assegurar que o ID do usuário e senha corretos sejam transmitidos para o servidor BioRS, crie mapeamentos de usuário em seu banco de dados federado para usuários que estão autorizados a procurar origens de dados BioRS. Quando criar um mapeamento de usuário, a senha remota é armazenada em um formato criptografado em uma tabela do catálogo do sistema do banco de dados federado.

Procedimento

Emita a instrução CRIAR MAPEAMENTO DE USUÁRIO a partir da linha de comandos. Por exemplo:

```
CREATE USER MAPPING FOR local_userID
SERVER server_definition_name
OPTIONS (REMOTE_AUTHID 'remote_userID',
REMOTE_PASSWORD 'remote_password')
PROXY_AUTHID 'proxy_server_userID',
PROXY_PASSWORD 'proxy_server_password';
```

Exemplo

Se especificar informações sobre autenticação do servidor proxy quando registrar uma definição de servidor e um mapeamento de usuário, os valores que você especificar na instrução CREATE USER MAPPING têm precedência sobre os valores especificados na instrução CREATE SERVER.

Por exemplo, você tem dez pessoas na sua organização e especifica informações sobre autenticação quando registra as definições de servidor. Você cria mapeamentos de usuário para três dessas dez pessoas. Quando as três pessoas acessarem o sistema federado, as informações sobre autenticação que você especificou quando criou os mapeamentos de usuário serão usadas. Para as outras sete pessoas, as informações sobre autenticação que você especificou quando registrou a definição do servidor serão usadas.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper BioRS:

Utilize a instrução CREATE USER MAPPING para mapear um ID do usuário do servidor federado para um ID do usuário e senha do servidor BioRS.

Você pode criar um mapeamento de usuário especificando uma conta de usuário guest, uma conta de usuário autenticado, o registro especial USER ou um servidor proxy.

Criando um Mapeamento de Usuário para uma Conta de Usuário Guest

A opção de usuário GUEST especifica se o wrapper BioRS deve utilizar uma conta guest para acessar o servidor BioRS.

O exemplo a seguir mostra como especificar que uma conta de usuário guest é utilizada para acessar o servidor BioRS:

```
CREATE USER MAPPING FOR charlie SERVER biors_server
OPTIONS (GUEST 'Y');
```

charlie Especifica o ID de autorização local que você está mapeando para um ID do usuário remoto, que está definido no servidor BioRS.

SERVER *biors_server*

Especifica o nome da definição do servidor registrado na instrução CREATE SERVER para o servidor BioRS.

GUEST 'Y'

Especifica que o wrapper BioRS utiliza uma conta de usuário guest para autenticar este usuário.

Criando um Mapeamento de Usuário para uma Conta de Usuário Autenticado

O exemplo a seguir mostra como mapear um ID do usuário do servidor federado para um ID do usuário e senha do servidor BioRS:

```
CREATE USER MAPPING FOR charlie SERVER biors_server
  OPTIONS (REMOTE_AUTHID 'charlene',
          REMOTE_PASSWORD 'all4one');
```

charlie Especifica o ID de autorização local que você está mapeando para um ID do usuário e senha remotos, que estão definidos no servidor BioRS.

SERVER *biors_server*

Especifica o nome da definição do servidor registrado na instrução CREATE SERVER para o servidor BioRS.

REMOTE_AUTHID '*charlene*'

Especifica o ID do usuário no servidor BioRS para o qual você está mapeando *charlie*. Este ID remoto deve estar em um formato esperado pelo servidor BioRS.

REMOTE_PASSWORD '*all4one*'

Especifica a senha que está associada a '*charlene*'.

Criando um Mapeamento de Usuário Utilizando um Registro Especial

Você pode utilizar o registro especial do banco de dados federado USER para mapear o ID de autorização da pessoa que está emitindo a instrução CREATE USER MAPPING para o ID de autorização da origem de dados especificado na opção do usuário REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER SERVER biors_server
  OPTIONS (REMOTE_AUTHID 'charlene', REMOTE_PASSWORD 'all4one');
```

Criando um Mapeamento de Usuário para um Servidor Proxy

O exemplo a seguir mostra como mapear um ID do usuário do servidor federado para um ID do usuário e senha do servidor BioRS:

Para registrar uma definição do servidor e especificar um servidor proxy SOCKS com informações sobre autenticação, utilize a seguinte instrução:

```
CREATE USER MAPPING FOR charlie SERVER biors_proxy
  OPTIONS (REMOTE_AUTHID 'charlene',
          REMOTE_PASSWORD 'all4one'
          PROXY_AUTHID 'chuck'
          PROXY_PASSWORD 'them2us');
```

PROXY_AUTHID '*chuck*'

Especifica o ID do usuário no servidor proxy. Esta opção de mapeamento de usuário é necessária quando o servidor proxy exige autenticação.

PROXY_PASSWORD '*them2us*'

Especifica a senha no servidor proxy que está associada ao nome de usuário '*chuck*'. Esta opção de mapeamento de usuário é necessária quando o servidor proxy exige autenticação.

Registrando Apelidos para Origens de Dados BioRS

Para cada definição do servidor BioRS registrada, você deve registrar um apelido para cada banco de dados que deseja acessar. Utilize estes apelidos, em vez dos nomes dos bancos de dados, quando consultar os servidores BioRS.

Antes de Iniciar

- Se um nome de banco de dados BioRS não estiver em conformidade com a sintaxe requerida pela instrução CREATE NICKNAME, será necessário utilizar a opção de apelido REMOTE_OBJECT quando registrar o apelido.
- Se um nome de elemento BioRS não estiver em conformidade com a sintaxe requerida pela instrução CREATE NICKNAME, será necessário utilizar a opção de coluna ELEMENT_NAME quando registrar o apelido.

Sobre Esta Tarefa

Restrições

Não utilize o elemento AllText do BioRS como a primeira coluna para um apelido. Você pode utilizar o elemento AllText do BioRS em qualquer outra posição de coluna (por exemplo, como a segunda coluna ou a terceira coluna).

Quando uma origem de dados tiver sido integrada ao sistema BioRS, ela será referida como um *banco de dados* no BioRS. Os bancos de dados no BioRS equivalem a apelidos em um sistema federado.

Os nomes que você dá aos apelidos podem ter até 128 caracteres.

Procedimento

Emita a instrução CRIAR APELIDO a partir da linha de comandos.

Por Exemplo:

```
CREATE NICKNAME nickname
(
  column_name data_type
  OPTIONS (nickname_column_options),
  column_name data_type
  OPTIONS (nickname_column_options),
  column_name data_type
  OPTIONS (nickname_column_options)
)
FOR SERVER server_definition_name
OPTIONS (nickname_options);
```

Quando criar um apelido BioRS, você define uma lista de colunas de apelidos. As colunas de apelidos especificadas devem corresponder aos elementos de um formato de banco de dados BioRS específico. O BioRS define cinco possíveis tipos de dados para elementos: Text, Number, Date, Author e Reference. Os tipos de dados BioRS podem ser mapeados apenas para os tipos de dados CHAR, CLOB ou VARCHAR que são utilizados pelo banco de dados federado.

O que Fazer Depois

Repita esta etapa para cada banco de dados BioRS para o qual deseja criar um apelido.

Instrução CREATE NICKNAME - Exemplos para o Wrapper BioRS:

Utilize a instrução CREATE NICKNAME para registrar um apelido para um banco de dados BioRS que você deseja acessar.

Criando Apelidos Simples mas Limitados para Bancos de Dados

A maneira mais simples de registrar um apelido para um banco de dados BioRS é fornecer ao apelido o mesmo nome que o banco de dados BioRS.

Por Exemplo:

```
CREATE NICKNAME SwissProt
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR(128),
  ENTRYDATE VARCHAR (64))
  FOR SERVER biors_server;
```

O nome do apelido é *SwissProt*, que é igual ao nome do banco de dados BioRS correspondente.

A utilização desta sintaxe simples de CREATE NICKNAME limita-o de duas maneiras:

1. Você está limitado a uma família de apelidos para cada esquema do banco de dados federado. Por exemplo, você tem dois bancos de dados que possuem um relacionamento pai-filho. Os bancos de dados são SWISSPROT e SPFEAT. Estes bancos de dados formam uma família. Se você utilizar a sintaxe padrão para a instrução CREATE NICKNAME, terá o (apelido SWISSPROT para o banco de dados SWISSPROT e o apelido SPFEAT para o banco de dados SPFEAT. Para ter mais de um apelido para SWISSPROT no esquema, é necessário utilizar a opção de apelido REMOTE_OBJECT ao registrar o apelido.
2. Você está limitado a bancos de dados cujos nomes podem ser utilizados como nomes de apelidos. O nome do banco de dados deve estar em conformidade com a sintaxe suportada pelo servidor federado. Por exemplo, se o nome do banco de dados contiver um ponto ou espaço, será necessário utilizar a opção de apelido REMOTE_OBJECT ao registrar o apelido.

Criando Vários Apelidos para o Mesmo Banco de Dados

A opção de apelido REMOTE_OBJECT especifica o nome do banco de dados BioRS que está associado ao apelido. O nome especificado na opção de apelido REMOTE_OBJECT determina o esquema e o banco de dados BioRS para o apelido. A opção de apelido REMOTE_OBJECT também especifica o relacionamento do apelido com outros apelidos.

O exemplo a seguir mostra o mesmo conjunto de características do apelido que o exemplo anterior, mas altera o nome do apelido. O exemplo utiliza a opção de apelido REMOTE_OBJECT para especificar o banco de dados BioRS para o qual o apelido está sendo definido:

```
CREATE NICKNAME NewSP
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR(128),
  ENTRYDATE VARCHAR (64))
  FOR SERVER biors_server
  OPTIONS (REMOTE_OBJECT 'SwissProt');
```

Criando Apelidos para Bancos de Dados que não Estão em Conformidade com a Sintaxe Federada

O exemplo a seguir mostra como criar um apelido para um banco de dados BioRS remoto que não está em conformidade com a sintaxe requerida pelo servidor federado:

```
CREATE NICKNAME SwissFT
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR (128),
  ENTRYDATE VARCHAR (64),
  FtLength VARCHAR (16))
  FOR SERVER biors1
  OPTIONS (REMOTE_OBJECT 'SwissProt.Features');
```

SwissFT

Um apelido exclusivo utilizado para identificar o banco de dados BioRS.

ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_')

O nome e tipo de dados para uma coluna da tabela. A opção de coluna `ELEMENT_NAME` é especificada para a coluna de ID.

A opção de coluna `ELEMENT_NAME` especifica o nome do elemento BioRS. A distinção entre maiúsculas e minúsculas deste nome depende da distinção entre maiúsculas e minúsculas do servidor BioRS e do valor da opção `CASE_SENSITIVE` do servidor. É necessário especificar o nome do elemento BioRS apenas se ele for diferente do nome da coluna. Os valores de opções de coluna devem ser colocados entre aspas simples.

Em geral, utilize a opção de coluna `ELEMENT_NAME` nas seguintes circunstâncias:

- Quando um nome de elemento BioRS contiver caracteres, como pontos e espaços, que não estão em conformidade com a sintaxe federada válida. Por exemplo, se seu banco de dados tiver um elemento denominado `Pub.Date`, não será possível utilizar o nome do elemento como o nome da coluna. Caracteres como pontos e espaços não são suportados. É necessário mapear o nome do elemento para um nome de coluna válido.
- Quando a sintaxe de um nome de elemento BioRS não estiver em conformidade com os padrões estabelecidos por você ou sua organização para seu sistema federado. Por exemplo, se sua organização tiver estabelecido que as convenções para esquemas, apelidos e colunas devem incluir um prefixo, um nome de elemento BioRS não poderá ser utilizado como um nome de coluna.
- Quando o nome do elemento BioRS não puder ser óbvio para usuários federados.

ALLTEXT VARCHAR(128)

O nome e tipo de dados para uma coluna da tabela.

ENTRYDATE VARCHAR(64)

O nome e tipo de dados para uma coluna da tabela.

FtLength VARCHAR(16)

O nome e tipo de dados para uma coluna da tabela.

SERVER *biors1*

O nome designado ao servidor BioRS na instrução `CREATE SERVER`.

OPTIONS (REMOTE_OBJECT '*SwissProt.Features*')

Especifica o nome do databank do BioRS que está associado ao

pseudônimo. Esse nome determina o esquema e o banco de dados do BioRS para o apelido. Esse nome também especifica o relacionamento do apelido com outros apelidos.

A distinção entre maiúsculas e minúsculas deste nome depende da distinção entre maiúsculas e minúsculas do servidor BioRS e do valor da opção CASE_SENSITIVE do servidor.

Importante: Não é possível alterar ou excluir este nome com a instrução ALTER NICKNAME. Se o nome do banco de dados BioRS for alterado, será necessário excluir e, em seguida, criar o apelido novamente.

Você deve especificar a opção de apelido REMOTE_OBJECT quando o nome de um banco de dados BioRS não estiver em conformidade com a sintaxe federada válida. Neste exemplo, o nome do banco de dados "SwissProt.Features" não está em conformidade por várias razões. O nome do banco de dados contém um caractere, um ponto, que não é uma sintaxe federada válida e contém uma combinação de letras maiúsculas e minúsculas.

Em geral, utilize a opção de apelido REMOTE_OBJECT nas seguintes circunstâncias:

- Quando um nome de banco de dados BioRS contiver caracteres, como pontos e espaços, que não estão em conformidade com a sintaxe federada válida. É necessário mapear o nome do banco de dados para um nome federado válido.
- Quando a distinção entre maiúsculas e minúsculas de um nome de banco de dados BioRS não estiver em conformidade com os padrões estabelecidos por você ou sua organização para seu sistema federado. Por exemplo, se sua organização tiver estabelecido que as convenções para esquemas, apelidos e colunas devem incluir um prefixo, um nome de banco de dados BioRS não poderá ser utilizado como um nome.
- Quando o nome do banco de dados BioRS não puder ser óbvio para usuários federados.

Criando Apelidos para um Banco de Dados Vinculado a outro Banco de Dados BioRS

O exemplo a seguir mostra como criar um apelido para uma tabela que utiliza um banco de dados BioRS que está vinculado a outro banco de dados BioRS:

```
CREATE NICKNAME SwissFT2
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR (1200),
  FtKey VARCHAR (32),
  FtLength VARCHAR (64),
  FtDescription VARCHAR (128),
  Parent VARCHAR (32) OPTIONS (REFERENCED_OBJECT 'SwissProt'))
FOR SERVER biors1
OPTIONS (REMOTE_OBJECT 'SwissProt.Features');
```

O nome deste apelido é SwissFT2. As colunas da tabela são ID, ALLTEXT, FtKey, FtLength, FtDescription e Parent. A opção de coluna ELEMENT_NAME é especificada para a coluna de ID. A opção REMOTE_OBJECT é utilizada para especificar o nome do banco de dados BioRS ao qual o apelido corresponde.

Além disso, a coluna Parent utiliza a opção REFERENCED_OBJECT. É necessário especificar esta opção para colunas que correspondem aos elementos de tipo de dados Reference do BioRS. A opção REFERENCED_OBJECT especifica o nome do

banco de dados BioRS ao qual a coluna se refere. Neste caso, o elemento Parent refere-se ao banco de dados BioRS SwissProt.

Funções Customizadas e Consultas BioRS

As funções customizadas BioRS são utilizadas com o wrapper BioRS para transferir predicados para o mecanismo de consulta BioRS.

O ambiente federado utiliza mecanismos de consulta. Para o wrapper BioRS, estes mecanismos de consulta são o banco de dados federado e o BioRS. Você pode especificar que predicados sejam transferidos para o mecanismo BioRS utilizando as quatro funções customizadas BioRS.

Todas as funções customizadas para o wrapper BioRS devem ser registradas com o nome do esquema BIORs. Você deve incluir o esquema BIORs sempre que utilizar as funções.

As funções customizadas para o wrapper BIORs são:

- BIORs.CONTAINS
- BIORs.CONTAINS_LE
- BIORs.CONTAINS_GE
- BIORs.SEARCH_TERM

As Funções BioRS CONTAINS

As funções customizadas BIORs.CONTAINS, BIORs.CONTAINS_LE e BIORs.CONTAINS_GE requerem um argumento de coluna do termo de procura e um argumento de texto de consulta. O exemplo a seguir mostra uma instrução BIORs.CONTAINS:

```
BIORs.CONTAINS (search_term_column,query_term)
```

O valor do argumento da coluna do termo de procura deve referir-se a uma coluna BioRS indexada. A utilização de uma coluna não indexada produz uma mensagem de erro SQL30090N ("Operação inválida para o ambiente de execução do aplicativo").

O valor do argumento do termo de consulta é o valor utilizado para procurar o elemento indexado especificado no argumento da coluna do termo de procura.

O valor do argumento do termo de consulta pode ser um literal, uma variável de host ou uma referência de coluna. Não é possível utilizar concatenação aritmética ou de cadeia. Além disso, o valor do argumento do termo de consulta não pode ser NULL, mesmo que a coluna do termo de procura utilizada esteja definida como permitindo valores nulos.

A distinção entre maiúsculas e minúsculas argumento do termo de consulta não é relevante.

Tipos de Dados Válidos

Os tipos de dados e formatos válidos do argumento do termo de consulta dependem do tipo de dados BioRS da coluna do termo de procura utilizada. O BioRS define cinco possíveis tipos de dados: Text, Author, Date, Number e Reference.

Os tipos de dados BioRS e os termos de consulta da função válidos para cada tipo de dados estão listados na tabela a seguir.

Tabela 10. Tipos de dados BioRS e termos de consulta de função customizada válidos

Tipo de dados da coluna do termo de procura	Termo de consulta válido	Formato
Texto	VARCHAR() ou CHAR()	Termo de texto do BioRS, incluindo curingas.
Author	VARCHAR() ou CHAR()	Referência do autor de BioRS no formato "<last>, <init>". "<last>" é o sobrenome do autor. "<init>" são as iniciais do autor, sem pontos. É permitido um espaço em branco entre a vírgula e as iniciais. Como alternativa, <last> pode ser especificado isoladamente, sem a vírgula ou iniciais.
Date	VARCHAR(), CHAR(), DATE ou TIMESTAMP	Se uma cadeia de caracteres, formato de data válido para o banco de dados federado, aaaa/mm/dd.
Número	VARCHAR() ou CHAR(), INTEGER, SMALLINT, BIGINT REAL, DOUBLE, DECIMAL	Formatos de número válidos para o banco de dados federado.
Referência	VARCHAR() ou CHAR()	Termo de texto do BioRS.

Todas as demais combinações de colunas de termo de procura e argumentos de termo de consulta do tipo de dados BioRS produzem a mensagem de erro SQL30090N ("Operação inválida para o ambiente de execução do aplicativo").

Utilizando Caracteres Curinga

O argumento de termo de consulta para colunas de termo de procura de tipo de dados Text, Author e Reference deve corresponder a um padrão de linguagem de consulta BioRS. No BioRS, os argumentos de termo de consulta podem consistir em cadeias alfanuméricas e curingas. A função BIOR.S.CONTAINS suporta dois curingas: ? (ponto de interrogação) e * (asterisco).

O curinga ? corresponde a um único caractere. Por exemplo, o predicado BIOR.S.CONTAINS (description, 'bacteri?')=1 corresponde ao termo bactéria mas não ao termo bacteriano.

O caractere curinga * corresponde a zero ou mais caracteres. Por exemplo, o predicado BIOR.S.CONTAINS (description, 'bacteri*')=1 corresponde aos termos bacteriologia, bactéria e bacteriano.

Para obter informações detalhadas sobre os padrões de linguagem de consulta BioRS, consulte a documentação do BioRS.

Especificando Funções BioRS CONTAINS em Consultas

A função BIOR.S.CONTAINS pode ser especificada para todos os tipos de colunas do BioRS.

As funções customizadas `BIORS.CONTAINS_GE` e `BIORS.CONTAINS_LE` podem ser especificadas apenas para colunas cujo tipo de dados BioRS subjacente seja `Number` ou `Date`. A função `BIORS.CONTAINS_GE` seleciona linhas nas quais a coluna contém um valor maior ou igual ao valor representado pelo argumento de termo de consulta. A função `BIORS.CONTAINS_LE` seleciona linhas nas quais a coluna contém um valor menor ou igual ao valor representado pelo argumento do termo de consulta.

As funções `BIORS.CONTAINS`, `BIORS.CONTAINS_GE` e `BIORS.CONTAINS_LE` retornam um resultado inteiro. Quando qualquer uma das três funções `CONTAINS` for utilizada em um predicado, o valor de retorno deverá ser comparado com o valor 1 utilizando os operadores `=` ou `<>`. Por exemplo:

```
SELECT * FROM s.MySP WHERE BIORS.CONTAINS (s.AllText, 'muscus') = 1;
```

A expressão `NOT (BioRS.Contains (col,value) = 1)` é equivalente à expressão `BioRS.CONTAINS (col,value) <> 1`.

A Função BioRS SEARCH_TERM

A função customizada `BIORS.SEARCH_TERM` requer um argumento de coluna do termo de procura e um termo de consulta. O exemplo a seguir mostra a sintaxe para a função customizada `SEARCH_TERM`:

```
BIORS.SEARCH_TERM (search_term_column,query_term)
```

O valor do argumento da coluna do termo de procura deve referir-se à coluna que representa o elemento `_ID_`.

O valor do argumento do termo de consulta é uma expressão que pode fazer referência a vários elementos.

O valor do argumento do termo de consulta pode ser um literal, uma variável de host ou uma referência de coluna. Não é possível utilizar concatenação aritmética ou de cadeia. Além disso, o valor do argumento do termo de consulta não pode ser `NULL`, mesmo que a coluna do termo de procura utilizada esteja definida como permitindo valores nulos.

A distinção entre maiúsculas e minúsculas argumento do termo de consulta não é relevante.

Especificando a Função BioRS SEARCH_TERM em Consultas

Você pode executar consultas que, de outra maneira, não seria possível, emitindo a função `BIORS.SEARCH_TERM`. Você pode utilizar esta função para especificar um termo de procura utilizando o formato BioRS. A função `BIORS.SEARCH_TERM` requer dois argumentos. O primeiro argumento é uma referência à coluna `_ID_` do apelido ao qual o termo deve ser aplicado. O segundo argumento é uma cadeia de caracteres que contém o termo sem um nome de banco de dados.

O exemplo a seguir seleciona todas as colunas para as entradas no banco de dados `MyEMBL` no qual o elemento `SeqLength` contém um valor maior ou igual a 100.

```
SELECT * FROM MyEMBL s WHERE  
  BIORS.SEARCH_TERM (s.ID, '[SeqLength GREATER number:100;]') = 1;
```

O exemplo a seguir seleciona a coluna `MolWeight` do apelido `Swiss` no qual o valor do elemento `MolWeight` é maior ou igual a 100368.

```
SELECT s.molweight FROM Swiss s WHERE
      BIoRS.SEARCH_TERM (s.ID, '[MolWeight GREATER number:100368;]') = 1;
```

Operações de Igualdade em Consultas BioRS

Você pode utilizar um operador de igualdade (=) em expressões literais ou em consultas de junção, com algumas limitações.

Se você utilizar o operador de igualdade em uma expressão literal ou em uma consulta de junção, o operador de igualdade deverá referir-se ao elemento `_ID_` de um banco de dados BioRS para que a consulta seja transferida para o servidor BioRS. As consultas que incluem um operador de igualdade mas que não fazem referência ao elemento `_ID_` não são transferidas para processamento pelo servidor BioRS.

Você pode utilizar o operador de igualdade em uma expressão literal. Por exemplo:

```
ID = 'swissprot:100K_RAT'
```

Você pode utilizar um predicado de igualdade em uma junção entre um banco de dados BioRS e outra tabela local ou apelidos não-BioRS. Por Exemplo:

```
SELECT n.ID, n.EntryDate, t.C1 FROM w46851_n1 n, w46851_t1 t WHERE t.ID = n.ID
```

Uma junção entre bancos de dados BioRS deve fazer referência ao elemento `_ID_` de um banco de dados e a um tipo de elemento de referência para o outro banco de dados.

No entanto, a utilização de um predicado de igualdade pode retornar resultados que se diferem dos resultados esperados nestas condições:

Correspondência sem distinção entre maiúsculas e minúsculas

A operação não faz distinção entre maiúsculas e minúsculas. Por exemplo, `ID='100k_rat'` corresponde às duas cadeias a seguir:

- '100k_rat'
- '100K_RAT'

Correspondência de curinga

A instrução `ID='100K_R*'` corresponde a '100K_RAT' e a '100K_RODENT.'

Prefixo do banco de dados

A operação retorna um prefixo que indica o banco de dados de origem. Por exemplo, `ID='100K_RAT'` em uma junção no banco de dados SwissProt pode retornar um valor de 'swissprot:100K_RAT.'

Nota: Não crie aplicativos que dependam de qualquer um dos comportamentos descritos.

O exemplo a seguir ilustra o comportamento do predicado de igualdade em uma junção.

A tabela local, `w46851_t1`, contém os seguintes valores:

ID	C1
swissprot:100K_RAT	0
swissprot:RAT	1
swissprot:100K_R	2
swissprot:100K_R*	3
swissprot:100k_rat	4
100K_RAT	100
RAT	101

100K_R	102
100K_R*	103
100k_rat	104

Você pode juntar a tabela w46851_t1 com um apelido w46851_n1 baseado no banco de dados SwissProt. A instrução a seguir mostra a consulta de junção com uma operação de igualdade:

```
SELECT n.ID, n.EntryDate, t.C1 FROM w46851_n1 n, w46851_t1 t WHERE t.ID = n.ID
```

São retornados os seguintes resultados:

ID	ENTRYDATE	C1
swissprot:100K_RAT	01-NOV-1997	0
swissprot:100K_RAT	01-NOV-1997	3
swissprot:100K_RAT	01-NOV-1997	4
swissprot:100K_RAT	01-NOV-1997	100
swissprot:100K_RAT	01-NOV-1997	103
swissprot:100K_RAT	01-NOV-1997	104

6 record(s) selected.

No entanto, o comportamento esperado é que apenas 0 linhas sejam retornadas.

Predicados de Junção Equivalente para o Wrapper BioRS

Você deve especificar predicados para o mecanismo do BioRS quando utilizar as funções customizadas BioRS, com uma exceção. A exceção é quando você desempenha operações de junção equivalente durante uma consulta.

Uma operação de *junção* envolve a recuperação de dados de duas ou mais tabelas com base em valores da coluna correspondentes. Uma *junção equivalente* é uma operação de junção na qual a condição de junção possui o formato expressão = expressão. Para consultas BioRS, os termos de junção equivalente devem conter o elemento `_ID_` de um banco de dados e um elemento de tipo Reference de outro banco de dados.

Exemplo para Definições de Apelido e uma Consulta de Junção Equivalente

Este exemplo mostra definições de apelido de amostra e uma consulta de junção equivalente que utiliza os apelidos de amostra.

Você deseja consultar dois bancos de dados BioRS, SwissProt e SwissProt.features. O banco de dados SwissProt.features é um filho do banco de dados SwissProt e contém um elemento chamado Parent. O elemento Parent contém referências a entradas identificadas pelo elemento `_ID_` de SwissProt. Você registra duas definições de apelido para os dois bancos de dados.

Definição de Apelido para o Banco de Dados SwissProt

O banco de dados SwissProt é o banco de dados pai.

```
CREATE NICKNAME tc600sprot (
  ID          VARCHAR (32) OPTIONS (ELEMENT_NAME '_ID_'),
  AllText     VARCHAR (128),
  EntryDate   VARCHAR (128),
  Update      VARCHAR (128),
  Description VARCHAR (1200),
  Crossreference VARCHAR (32),
  Authors     VARCHAR (256),
  Journal     VARCHAR (256),
```

```

JournalIssue    VARCHAR (64) OPTIONS (IS_INDEXED 'N'),
PublicationYear VARCHAR (1024),
Gene            VARCHAR (20) OPTIONS (IS_INDEXED 'Y'),
Remarks        VARCHAR (1200),
RemarkType      CHAR (20),
CatalyticActivity VARCHAR (20),
CoFactor        VARCHAR (64),
Disease         VARCHAR (128),
Function        VARCHAR (128),
Pathway         VARCHAR (128),
Similarity      VARCHAR (128),
Complex         VARCHAR (64),
FtKey           VARCHAR (32),
FtDescription   VARCHAR (128),
FtLength        VARCHAR (256),
MolWeight       VARCHAR (64),
ProteinLen      VARCHAR (32) OPTIONS (ELEMENT_NAME 'Protein_length'),
Sequence        CLOB,
AccNumber       VARCHAR (32),
Taxonomy        VARCHAR (128),
Organelle       VARCHAR (128),
Organism        VARCHAR (128),
Keywords        VARCHAR (1200),
Localization    VARCHAR (128),
FtKey_count     VARCHAR (32))
FOR SERVER biors_server_600
  OPTIONS (REMOTE_OBJECT 'SwissProt');

```

Definição de Apelido para o Banco de Dados SwissProt.features

O banco de dados SwissProt.features é um banco de dados filho do banco de dados SwissProt. Este apelido contém o elemento **Parent**.

```

CREATE NICKNAME tc600feat (
  ID            VARCHAR (32) OPTIONS (ELEMENT_NAME '_ID_'),
  AllText       VARCHAR (1200),
  FtKey         VARCHAR (32),
  FtLength      VARCHAR (64),
  FtDescription VARCHAR (128),
  Parent        VARCHAR (32) OPTIONS (REFERENCED_OBJECT 'SwissProt'))
FOR SERVER biors_server_600
  OPTIONS (REMOTE_OBJECT 'SwissProt.features');

```

A Consulta que Faz Referência a Dois Apelidos em uma Junção Equivalente

Nesta consulta, dois predicados são aplicados ao apelido tc600sprot (banco de dados SwissProt). Estes dois predicados filtram as linhas que contêm o termo anopheles e têm um ano de publicação de 1997. Um predicado é aplicado ao apelido tc600feat (banco de dados SwissProt.features), que filtra as linhas cujo elemento FtKey contém o termo signal. Os dois apelidos são juntados utilizando o termo $f.Parent = s.ID$.

```

SELECT s.ID, f.ID, f.FtKey FROM tc600sprot s, tc600feat f
  WHERE BIORS.CONTAINS (s.AllText, 'anopheles') = 1
  AND BIORS.CONTAINS (s.PublicationYear, 1997) = 1
  AND BIORS.CONTAINS (f.FtKey, 'signal') = 1
  AND f.Parent = s.ID;

```

O conjunto de resultados final contém apenas as linhas que atendem estes critérios e as entradas no banco de dados SwissProt.features que fazem referência a uma entrada correspondente no banco de dados SwissProt.

O Elemento AllText do BioRS

Cada banco de dados no sistema BioRS contém um elemento chamado AllText. O elemento AllText é um elemento indexado criado automaticamente pelo BioRS para todos os bancos de dados.

Utilizando o elemento AllText, você pode procurar todo o texto em uma entrada, não apenas em elementos indexados específicos. Por exemplo, procurar o termo *musculus* pode retornar entradas nas quais a palavra *musculus* aparece no título, abstrato, descrição ou organismo.

Para utilizar o elemento AllText em uma consulta federada, é necessário mapear o elemento AllText para uma coluna de apelido. Você mapeia o elemento AllText para uma coluna de apelido quando especifica colunas na instrução CREATE NICKNAME. Uma coluna de apelido mapeada para o elemento AllText retorna um valor NULL em instruções SELECT. Ao especificar uma coluna como um elemento AllText, a coluna não deverá ser a primeira coluna declarada em uma instrução CREATE NICKNAME.

Quando o elemento AllText for mapeado corretamente para uma coluna de apelido, será possível utilizar essa coluna de apelido em uma função customizada BIOR.S.CONTAINS.

Origem de Dados BioRS - Consultas de Exemplo

Estes exemplos incluem um conjunto abrangente de consultas de amostra que podem ser utilizadas para acessar origens de dados BioRS e mostrar as instruções necessárias para criar os apelidos utilizados nos exemplos.

Estes dois exemplos mostram como:

- Estruturar suas consultas para otimizar o desempenho do sistema
- Utilizar as funções customizadas e curingas em suas consultas
- Utilizar consultas para acessar colunas de tipos de dados BioRS específicas
- Utilizar predicados relacionados para formar uma junção equivalente entre os apelidos pai e filho

As consultas de amostra utilizam os apelidos *swiss* e *swissft*.

Instrução CREATE NICKNAME para o Apelido *swiss*

O apelido pai *swiss* foi registrado no banco de dados SwissProt utilizando a seguinte instrução CREATE NICKNAME:

```
CREATE NICKNAME swiss
(
  ID                CHAR (30) OPTIONS (ELEMENT_NAME '_ID_'),
  EntryDate         VARCHAR (15),
  Update            CLOB (15),
  Description        CLOB (15),
  Crossreference     CLOB (15),
  Authors            CLOB (15),
  Journal            VARCHAR (15),
  JournalIssue       VARCHAR (15),
  PublicationYear    CLOB (15),
  PublicationTitle   CLOB (15),
  Gene               CLOB (15),
  Remarks            CLOB (15),
  RemarkType         VARCHAR (15),
  CatalyticActivity VARCHAR (15),
  CoFactor           VARCHAR (15),
  Disease            VARCHAR (15),
```

```

Function          CLOB (15),
Pathway           VARCHAR (15),
Similarity        CLOB (15),
Complex           VARCHAR (15),
FtKey             VARCHAR (15),
FtDescription     CLOB (15),
FtLength          VARCHAR (15),
MolWeight         CHAR (15),
Protein_Length   VARCHAR (15),
Sequence          CLOB (15),
AccNumber         VARCHAR (15),
Taxonomy          CLOB (15),
Organelle         VARCHAR (15),
Organism          VARCHAR (15),
Keywords          VARCHAR (15),
Localization      VARCHAR (15),
FtKey_count       VARCHAR (15),
AllText           CLOB (15)
)
FOR SERVER biors_server
  OPTIONS (REMOTE_OBJECT 'swissprot');

```

Instrução CREATE NICKNAME para o Apelido swissft

O apelido filho swissft foi registrado para o banco de dados SwissProt.Features utilizando a seguinte instrução CREATE NICKNAME:

```

CREATE NICKNAME swissft
(
  ID          VARCHAR (30) OPTIONS (ELEMENT_NAME '_ID_'),
  FtKey       VARCHAR (15),
  FtLength    VARCHAR (15),
  FtDescription VARCHAR (15),
  Parent      VARCHAR (30) OPTIONS (REFERENCED_OBJECT 'swissprot'),
  AllText     CLOB (15)
)
FOR SERVER biors_server
  OPTIONS (REMOTE_OBJECT 'swissprot.features');

```

Estruturas de Consultas Impactam o Desempenho do Servidor Federado

As consultas e resultados na tabela a seguir mostram como é possível estruturar suas consultas para otimizar a carga de trabalho entre o sistema federado e o servidor BioRS.

Tabela 11. Amostras de Diferentes Consultas que Produzem Resultados Idênticos

Consulta	Resultado
SELECT s.id FROM Swiss s WHERE BIORS.CONTAINS(s.id, '100K_RAT') = 1 FETCH FIRST 3 ROWS ONLY;	ID ----- 100K_RAT 1 record(s) selected.
SELECT s.id FROM Swiss s WHERE s.id LIKE '%100K_RAT%' FETCH FIRST 3 ROWS ONLY;	ID ----- 100K_RAT 1 record(s) selected.

Duas das consultas nesta tabela produzem os mesmos resultados. No entanto, a primeira consulta é executada mais rapidamente do que a segunda. A primeira

consulta utiliza a função BIOR.S.CONTAINS para especificar o predicado de entrada. Portanto, o servidor BioRS seleciona os dados no banco de dados SwissProt, em seguida, transmite os dados selecionados para o servidor federado. Na segunda consulta, o predicado de entrada LIKE é especificado diretamente no apelido swiss. Portanto, o servidor BioRS transfere todo o banco de dados SwissProt para o servidor federado. Quando o conteúdo do banco de dados for transferido, o servidor federado então selecionará os dados.

O desempenho da consulta geralmente é muito melhor quando os predicados são enviados para a origem de dados para processamento.

Consultas que Utilizam Curingas na Função Customizada BIOR.S.CONTAINS

As consultas e resultados na tabela a seguir mostram a utilização de caracteres curinga na função customizada BIOR.S.CONTAINS. Todos os resultados da consulta são idênticos, mesmo que sejam utilizados diferentes caracteres curinga.

Tabela 12. Consultas de Amostra que Utilizam Curingas na Função Customizada BIOR.S.CONTAINS

Consulta	Resultado
SELECT s.crossreference FROM Swiss s WHERE BIOR.S.CONTAINS(s.crossreference, 'MEDLINE') = 1 FETCH FIRST 3 ROWS ONLY;	CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081 3 record(s) selected.
SELECT s.crossreference FROM Swiss s WHERE BIOR.S.CONTAINS(s.crossreference, '?ED?IN?') = 1 FETCH FIRST 3 ROWS ONLY;	CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081 3 record(s) selected.
SELECT s.crossreference FROM Swiss s WHERE BIOR.S.CONTAINS(s.crossreference, '*D*N*') = 1 FETCH FIRST 3 ROWS ONLY;	CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081 3 record(s) selected

Consultas que Acessam Colunas de Tipo de Dados BioRS Author

As consultas e resultados na tabela a seguir mostram como é possível acessar informações em elementos de tipo de dados BioRS Author com a função customizada BIOR.S.CONTAINS. A sintaxe de todas as consultas é quase idêntica. A única diferença é a presença ou ausência da primeira inicial no termo de consulta e a quantidade de espaço entre o nome e a última inicial.

Tabela 13. Consultas de Amostra que Acessam Colunas de Tipo de Dados BioRS Author

Consulta	Resultado
SELECT s.authors FROM Swiss s WHERE BIORS.CONTAINS(s.authors, 'Mueller') = 1 FETCH FIRST 3 ROWS ONLY;	AUTHORS ----- Mueller D. Rehb Mayer K.F.X. Sc Zemmour J. Litt 3 record(s) selected.
SELECT s.authors FROM Swiss s WHERE BIORS.CONTAINS(s.authors, 'Mueller,D') = 1 FETCH FIRST 3 ROWS ONLY;	AUTHORS ----- 0 record(s) selected.
SELECT s.authors FROM Swiss s WHERE BIORS.CONTAINS(s.authors, 'Mueller ,D') = 1 FETCH FIRST 3 ROWS ONLY;	AUTHORS ----- 0 record(s) selected.
SELECT s.authors FROM Swiss s WHERE BIORS.CONTAINS(s.authors, 'Mueller, D') = 1 FETCH FIRST 3 ROWS ONLY;	AUTHORS ----- Mueller D. Rehb Zou P.J. Borovo Davies J.D. Mue 3 record(s) selected.

Consultas que Acessam Colunas de Tipo de Dados BioRS Date

As consultas e resultados na tabela a seguir mostram como é possível acessar informações em elementos de tipo BioRS Date com a função customizada BIORS.CONTAINS.

Quando um campo de tipo BioRS Date contiver uma seqüência de datas, os resultados poderão conter informações extras, conforme mostrado no segundo exemplo na tabela. Os elementos de tipo de dados BioRS Numeric (Date e Number) podem conter vários valores. Portanto, os resultados das consultas executadas nos elementos BioRS Date ou Number também podem conter vários valores. Vários valores são sempre separados por espaços.

Tabela 14. Consultas de Amostra que Acessam Colunas de Tipo de Dados BioRS Date

Consulta	Resultado
SELECT e.entrydate FROM embl e WHERE BIORS.CONTAINS(e.entrydate, date('11/01/1997')) = 1 FETCH FIRST 3 ROWS ONLY;	ENTRYDATE ----- 01-NOV-1997 01-NOV-1997 01-NOV-1997 3 record(s) selected.
SELECT g.update FROM gen g WHERE BIORS.CONTAINS(g.update, date('11/01/1997')) = 1 FETCH FIRST 3 ROWS ONLY;	UPDATE ----- 01-NOV-1997 11- 01-NOV-1997 12- 01-NOV-1997 06- 3 record(s) selected.

Consultas que Utilizam as Funções Customizadas BIORS.CONTAINS_LE e BIOR.S.CONTAINS_GE

As consultas e resultados na tabela a seguir mostram como utilizar as funções customizadas BIOR.S.CONTAINS_LE e BIOR.S.CONTAINS_GE.

Tabela 15. Consultas de Amostra que Utilizam as Funções Customizadas BIOR.S.CONTAINS_LE e BIOR.S.CONTAINS_GE

Consulta	Resultado
SELECT s.molweight FROM Swiss s WHERE BIORS.CONTAINS_LE(s.molweight, 100368) = 1 FETCH FIRST 3 ROWS ONLY;	MOLWEIGHT ----- 100368 10576 8523 3 record(s) selected.
SELECT s.molweight FROM Swiss s WHERE BIORS.CONTAINS_GE(s.molweight, 100368) = 1 FETCH FIRST 3 ROWS ONLY;	MOLWEIGHT ----- 100368 103625 132801 3 record(s) selected.
SELECT s.journalissue FROM Swiss s WHERE BIORS.CONTAINS_GE(s.journalissue, 172) = 1 FETCH FIRST 3 ROWS ONLY;	JOURNALISSUE ----- 172 21 242 196 3 record(s) selected.

Consultas que Utilizam a Função Customizada BIOR.S.SEARCH_TERM

As consultas e resultados na tabela a seguir mostram como é possível utilizar a função customizada BIOR.S.SEARCH_TERM para especificar um termo de procura utilizando o formato BioRS.

Tabela 16. Consultas de Amostra que Utilizam a Função Customizada BIOR.S.SEARCH_TERM

Consulta	Resultado
SELECT s.publicationyear FROM Swiss s WHERE BIORS.SEARCH_TERM (s.id, '[PublicationYear EQ number:1997;]')=1 FETCH FIRST 10 ROWS ONLY;	PUBLICATIONYEAR ----- 1997 1997 2000 1988 1991 1997 1994 1997 1997 1998 1994 1995 1997 1997 1999 1997 1994 1994 1995 1993 1992 1997 10 record(s) selected.

Tabela 16. Consultas de Amostra que Utilizam a Função Customizada BIOR.SSEARCH_TERM (continuação)

Consulta	Resultado
SELECT s.molweight FROM Swiss s WHERE BIOR.SSEARCH_TERM (s.id, '[MolWeight EQ number:100368;]') = 1 FETCH FIRST 10 ROWS ONLY;	MOLWEIGHT ----- 100368 100368 2 record(s) selected.
SELECT s.molweight FROM Swiss s WHERE BIOR.SSEARCH_TERM (s.id, '[MolWeight GREATER number:100368;]') = 1 FETCH FIRST 10 ROWS ONLY;	MOLWEIGHT ----- 100368 103625 132801 194328 130277 287022 289130 135502 112715 112599 10 record(s) selected.

Utilizando Predicados Relacionais para Formar uma Junção Equivalente entre Dois Bancos de Dados que Possuem um Relacionamento Pai-filho

A consulta a seguir mostra como utilizar predicados relacionais para formar uma junção equivalente entre dois bancos de dados que possuem um relacionamento pai-filho:

```
SELECT s.id, f.id, f.parent FROM Swiss s, Swissft f
WHERE (f.parent = s.id) FETCH FIRST 10 ROWS ONLY;
```

Nos resultados da consulta a seguir, o registro 100K_RAT é um pai para nove registros filhos (100K_RAT.1 a 100K_RAT.9).

ID	ID	PARENT
100K_RAT	100K_RAT.1	swissprot:100K_RAT
100K_RAT	100K_RAT.2	swissprot:100K_RAT
100K_RAT	100K_RAT.3	swissprot:100K_RAT
100K_RAT	100K_RAT.4	swissprot:100K_RAT
100K_RAT	100K_RAT.5	swissprot:100K_RAT
100K_RAT	100K_RAT.6	swissprot:100K_RAT
100K_RAT	100K_RAT.7	swissprot:100K_RAT
100K_RAT	100K_RAT.8	swissprot:100K_RAT
100K_RAT	100K_RAT.9	swissprot:100K_RAT
104K_THEPA	104K_THEPA.1	swissprot:104K_THEPA

10 record(s) selected.

Otimizando o Desempenho do Wrapper BioRS

Você pode aprimorar o desempenho de consultas em origens de dados BioRS otimizando o desempenho do wrapper BioRS.

Orientações para Otimizar o Desempenho do Wrapper BioRS

A estrutura de suas consultas e informações estatísticas sobre os bancos de dados BioRS impactam o desempenho de consultas.

Minimize a quantidade de dados transferidos entre mecanismos de procura.

O ambiente federado utiliza dois mecanismos de procura. Para o wrapper BioRS, estes mecanismos de procura são o banco de dados federado e o BioRS. Algumas configurações incluem mais de um mecanismo de banco de dados federado. O mecanismo de banco de dados federado processa predicados (operadores relacionais, tais como, =, BETWEEN, LIKE e <>) especificados em colunas de apelidos. O mecanismo do BioRS processa predicados especificados utilizando quatro funções customizadas para o wrapper BioRS.

Para minimizar a quantidade de dados transferidos entre os dois mecanismos de procura, estruture suas consultas para que o processamento de dados seja transferido para o sistema BioRS sempre que possível.

Se precisar desempenhar operações de junção em uma consulta, tire vantagem de relacionamentos pai-filho já existentes nos bancos de dados BioRS e desempenhe operações de junção equivalente sempre que possível. As operações de junção equivalente são processadas no BioRS, que também minimiza a quantidade de dados transferidos entre o banco de dados federado e os mecanismos de procura do BioRS.

Importante: Não interrompa consultas federadas no BioRS, por exemplo, utilizando **Ctrl-D** ou **Ctrl-Z** no processador de linha de comandos ou parando um programa aplicativo. Interromper uma consulta deixa processos "inativos" em execução no servidor BioRS. Estes processos "inativos" afetarão rapidamente o desempenho do servidor BioRS e do servidor federado. Se houver muitos destes processos "inativos" em execução, poderão ocorrer erros inesperados durante o processamento de consulta federada. Por exemplo, uma consulta válida pode retornar 0 linhas, quando são esperadas linhas. Em situações extremas, o servidor BioRS, o servidor federado ou os dois servidores podem ser parados ou encerrados de forma anormal.

Mantenha informações estatísticas do BioRS no ambiente federado.

Em um sistema federado, o banco de dados federado depende de estatísticas do catálogo para objetos apelidados para otimizar o processamento de consultas. A manutenção de estatísticas atuais sobre as origens de dados BioRS é essencial para otimizar o desempenho do wrapper BioRS. Se os dados estatísticos ou características estruturais para um objeto remoto no qual um apelido está definido forem alteradas, será necessário atualizar as estatísticas de cardinalidade da coluna de apelido correspondente em seu sistema federado.

Para otimizar o desempenho do wrapper BioRS, desempenhe estas atualizações no servidor federado em intervalos regulares.

Informações Estatísticas do BioRS

As informações estatísticas atuais do BioRS são essenciais para otimizar o desempenho do wrapper BioRS.

Em um sistema federado, o banco de dados federado depende de estatísticas do catálogo para objetos com apelidos para otimizar o processamento de consultas. Estas estatísticas são recuperadas de origens de dados BioRS quando você registra um apelido utilizando a instrução CREATE NICKNAME. O banco de dados federado verifica a presença do objeto na origem de dados e, em seguida, tenta reunir dados estatísticos da origem de dados existentes. As informações são lidas a partir dos catálogos da origem de dados e colocadas no catálogo do sistema no banco de dados federado.

Para origens de dados BioRS, as informações estatísticas críticas incluem:

- A cardinalidade de um apelido. Para origens de dados BioRS, a cardinalidade do apelido é equivalente ao número de entradas no banco de dados BioRS correspondente.
- A cardinalidade da coluna que corresponde ao elemento `_ID_` do BioRS. A cardinalidade desta coluna deve corresponder à cardinalidade do apelido no qual a coluna é referida.
- A cardinalidade de todas as colunas que o wrapper BioRS pode precisar utilizar.

Dica: Para otimizar o desempenho do wrapper BioRS, você deve manter estatísticas atuais sobre as origens de dados BioRS. Se os dados estatísticos ou características estruturais forem alteradas para um objeto remoto no qual um apelido está definido, será necessário atualizar as estatísticas de cardinalidade correspondentes em seu sistema federado. As estatísticas de cardinalidade estão armazenadas no catálogo do sistema do banco de dados federado na visualização `SYSSTAT.TABLES` e na visualização `SYSSTAT.COLUMNS`.

Para manter estatísticas de cardinalidade do BioRS em seu sistema federado:

1. Determine as estatísticas de cardinalidade para o apelido, se necessário.
2. Atualize as estatísticas de cardinalidade para o apelido nas visualizações do catálogo.
3. Atualize as estatísticas de cardinalidade para as colunas nas visualizações do catálogo.

Determinando Estatísticas de Cardinalidade do Banco de Dados BioRS

Você deve determinar estatísticas de cardinalidade do banco de dados BioRS antes de atualizar as estatísticas de apelido ou de atualizar a cardinalidade da coluna que corresponde ao elemento `_ID_` do BioRS.

Procedimento

Utilize o programa utilitário `admin_find` ou o programa utilitário `www_find.cgi` do BioRS e especifique a opção `-c`, que é para cardinalidade. Para obter informações adicionais sobre estes dois programas utilitários do BioRS, consulte a documentação do BioRS.

Atualizando Estatísticas de Cardinalidade de Apelido BioRS

Você deve atualizar as estatísticas de cardinalidade de apelido quando o conteúdo de um banco de dados BioRS for alterado significativamente.

Antes de Iniciar

Você deve determinar o número de cardinalidade do banco de dados BioRS que corresponde ao apelido cujas estatísticas você deseja atualizar.

Sobre Esta Tarefa

Para atualizar estatísticas de cardinalidade de apelido BioRS em seu sistema federado, é necessário modificar a visualização de catálogo `SYSSTAT.TABLES`.

A manutenção das estatísticas de cardinalidade corretas para apelidos permite que o otimizador e o wrapper BioRS escolham o plano de acesso a dados com o melhor desempenho.

Procedimento

Emita a instrução UPDATE para modificar a visualização de catálogo SYSSTAT.TABLES e especificar o número de cardinalidade correto. A sintaxe da instrução UPDATE é:

```
UPDATE SYSSTAT.TABLES SET CARD=cardinality_number
      WHERE TABSCHEMA=nickname_schema
      AND TABNAME=nickname_name;
```

Por exemplo, se o apelido for JONES.SWISS, utilize a seguinte instrução UPDATE para atualizar as estatísticas:

```
UPDATE SYSSTAT.TABLES SET CARD=15312191
      WHERE TABSCHEMA='JONES'
      AND TABNAME='SWISS';
```

SYSSTAT.TABLES

A visualização de catálogo do sistema no banco de dados federado no qual as estatísticas de apelido estão armazenadas.

SET CARD=15312191

O número de cardinalidade do banco de dados BioRS que corresponde ao apelido para o qual você está atualizando as estatísticas.

TABSCHEMA= 'JONES'

O nome do esquema para o apelido que você deseja atualizar.

TABNAME='SWISS'

O nome do apelido que você deseja atualizar.

Atualizando Estatísticas de Cardinalidade de Coluna BioRS

Para atualizar estatísticas de cardinalidade de coluna BioRS em seu sistema federado, é necessário modificar a visualização de catálogo SYSSTAT.COLUMNS.

Sobre Esta Tarefa

Você pode atualizar estatísticas de cardinalidade de coluna BioRS antes de criar apelidos do BioRS ou de atualizar estatísticas de cardinalidade de coluna BioRS quando desejar aprimorar o desempenho de consultas para origens de dados BioRS.

Restrições

Não utilize este procedimento para atualizar as estatísticas de cardinalidade para colunas que correspondem ao elemento `_ID_` do BioRS.

Você deve assegurar que as estatísticas de cardinalidade para colunas BioRS sejam atuais para que o otimizador e o wrapper BioRS possam escolher o melhor plano de acesso a dados durante o processamento de consultas.

Procedimento

Emita a instrução UPDATE para modificar a visualização de catálogo SYSSTAT.COLUMNS. A sintaxe da instrução UPDATE é:

```
UPDATE SYSSTAT.COLUMNS SET COLCARD=(SELECT COUNT(DISTINCT column_name)
      FROM nickname_schema.nickname_name)
      WHERE
      TABSCHEMA=nickname_schema
      AND TABNAME=nickname_name
      AND COLNAME=column_name;
```

SYSSTAT.COLUMNS

A visualização de catálogo do sistema no banco de dados federado no qual as estatísticas de colunas estão armazenadas.

COLCARD=(SELECT COUNT(DISTINCT *column_name*

O nome da coluna no apelido para a qual você está atualizando as estatísticas.

TABSCHEMA= *nickname_schema*

O nome do esquema para o apelido que você deseja atualizar.

TABNAME=*nickname_name*

O nome do apelido que você deseja atualizar.

COLNAME=*column_name*

O nome da coluna cujas estatísticas de cardinalidade você deseja atualizar.

A execução da consulta pode levar alguns minutos, porque todas as entradas para o banco de dados associado ao apelido devem ser recuperadas.

Se uma coluna contiver vários valores (por exemplo, o elemento `PublicationYear` do formato do banco de dados `SwissProt`), o cálculo se tornará muito complexo para utilizar uma consulta SQL. Para tais colunas, é necessário calcular manualmente o valor de cardinalidade e, em seguida, atualizar a visualização de catálogo `SYSSTAT.COLUMNS`. Para calcular o valor de cardinalidade, divida o número de valores distintos na coluna pelo número médio de valores por linha. O valor de cardinalidade calculado não pode ser maior do que a cardinalidade da tabela.

Por exemplo, se o apelido tiver as três seguintes linhas de valores para a coluna `PublicationYear`, haverá nove valores distintos e o número médio de valores em uma linha será quatro.

- 1997 1992 1985
- 1997 1992 1982
- 1992 1991 1990 1976 1974 1971

A cardinalidade para esta coluna `PublicationYear` é 9 dividido por 4 ou 3 (2,25 arredondado para o próximo inteiro mais alto). Você pode atualizar a visualização de catálogo `SYSSTAT.COLUMNS` utilizando a seguinte instrução `UPDATE`:

```
UPDATE SYSSTAT.COLUMNS SET CARDCOL=3
WHERE
  TABSCHEMA=nickname_schema
  AND TABNAME=nickname_name
  AND COLNAME=column_name;
```

Atualizando a Cardinalidade de Coluna `_ID_` do BioRS

Para atualizar estatísticas de cardinalidade de coluna BioRS para a coluna mapeada para o elemento `_ID_` do BioRS, é necessário modificar a visualização de catálogo `SYSSTAT.COLUMNS`.

Antes de Iniciar

Você deve determinar o número de cardinalidade do banco de dados BioRS que corresponde ao apelido no qual a coluna é referida. O número de cardinalidade da coluna mapeada para o elemento `_ID_` do BioRS deve corresponder à cardinalidade do apelido no qual a coluna é referida.

Você deve assegurar que as estatísticas de cardinalidade para a coluna mapeada para o elemento `_ID_` do BioRS sejam atuais. O otimizador e o wrapper BioRS

utilizam estas estatísticas para escolher o melhor plano de acesso a dados para processar suas consultas.

Sobre Esta Tarefa

Para atualizar a cardinalidade de coluna `_ID_` do BioRS, é necessário selecionar entradas na visualização `SYSCAT.COLOPTIONS` que contenham a opção `ELEMENT_NAME`. Este wrapper BioRS utiliza esta opção para mapeamento entre nomes de colunas de apelido no banco de dados federado e nomes de elementos no servidor BioRS.

Procedimento

Emita a instrução `UPDATE` para modificar a visualização de catálogo.

Por Exemplo:

```
UPDATE SYSSTAT.COLUMNS SET COLCARD=cardinality_number
WHERE
  TABSCHEMA=nickname_schema
  AND TABNAME=nickname_name
  AND COLNAME=column_name
  IN (SELECT column_name FROM SYSCAT.COLOPTIONS
      WHERE
        TABSCHEMA=nickname_schema
        AND TABNAME=nickname_name
        AND OPTION='ELEMENT_NAME'
        AND SETTING='_ID_')
```

SYSSTAT.COLUMNS

A visualização de catálogo do sistema no banco de dados federado no qual as estatísticas de colunas estão armazenadas.

SET COLCARD=*cardinality_number*

O número de cardinalidade do banco de dados BioRS que corresponde ao apelido da coluna para a qual você está atualizando estatísticas.

TABSCHEMA= *nickname_schema*

O nome do esquema para o apelido que você deseja atualizar.

TABNAME=*nickname_name*

O nome do apelido que você deseja atualizar.

COLNAME=*column_name*

O nome da coluna cujas estatísticas de cardinalidade você deseja atualizar.

IN (SELECT *column_name* FROM SYSCAT.COLOPTIONS

Esta instrução `SELECT` determina o nome da coluna mapeada para o elemento `_ID_` do BioRS. `SYSCAT.COLOPTIONS` é a visualização de catálogo do sistema no banco de dados federado no qual as opções de colunas estão armazenadas.

OPTION='ELEMENT_NAME'

O valor nas linhas na visualização `SYSCAT.COLOPTIONS` que indica que um nome de coluna de apelido é mapeado para o nome do elemento BioRS.

SETTING='_ID_'

Especifica que a coluna para a opção `ELEMENT_NAME` é `'_ID_'`.

Configurando o Acesso às Origens de Dados do DB2

Para configurar um servidor federado para acessar as origens de dados da família DB2, você deve fornecer ao servidor federado informações sobre as origens de dados e os objetos que deseja acessar.

Antes de Iniciar

- Verifique a configuração do servidor federado.

Procedimento

1. Catalogue uma entrada do nó DB2.
2. Catalogue o banco de dados DB2 remoto.
3. Registre o wrapper DB2.
4. Registre as definições do servidor para uma origem de dados DB2.
5. Crie os mapeamentos de usuário para uma origem de dados DB2.
6. Teste a conexão para o servidor de origem de dados do DB2.
7. Registre os apelidos para as tabelas e visualizações do DB2.

Catalogando uma Entrada de Nó do DB2

Você deve catalogar uma entrada de nó que especifique o protocolo que o servidor federado usa para se conectar à origem de dados do DB2.

Procedimento

Emita o comando **CATALOG TCPIP NODE**.

Por Exemplo:

```
CATALOG TCPIP NODE db2_node REMOTE system42 SERVER db2tcp42
```

em que:

- *db2_node* é o nome que você designa ao nó.
- *system42* é o nome do host do sistema em que a origem de dados reside.
- *db2tcp42* é o nome do serviço ou número de porta primária da instância do gerenciador de banco de dados do servidor.

Catalogando o Banco de Dados DB2 Remoto

Para identificar a quais bancos de dados o servidor federado se conecta, você deve catalogar o banco de dados DB2 remoto no diretório do banco de dados do sistema do servidor federado.

Procedimento

1. Use o Assistente de Configuração para catalogar o banco de dados ou emita o comando **CATALOG DATABASE**.

Por exemplo:

```
CATALOG DATABASE DB2DB390 AS CLIENTS390 AT NODE DB2NODE AUTHENTICATION SERVER
```

em que:

- *DB2DB390* é o nome do banco de dados remoto a ser catalogado.
- *CLIENTS390* é o alias para o banco de dados remoto que está sendo catalogado. Se você não especificar um alias, o gerenciador de banco de dados usa o nome do banco de dados remoto como o alias.
- *DB2NODE* é o nome do nó que você catalogou anteriormente.

- AUTHENTICATION SERVER especifica que a autenticação acontece no nó da origem de dados do DB2.
2. Se o nome do banco de dados remoto tiver mais de oito caracteres, emita o comando **CATALOG DCS DATABASE** para criar uma entrada de diretório DCS.

Por Exemplo:

```
CATALOG DCS DATABASE SALES400 AS SALES_DB2DB400
```

em que:

- SALES400 é o alias do banco de dados remoto a ser catalogado. O alias deve corresponder ao nome de uma entrada no diretório do banco de dados do sistema do servidor federado que está associado ao nó remoto. O alias é o mesmo nome que você especificou no comando **CATALOG DATABASE**.
- SALES_DB2DB400 é o nome do banco de dados do host de destino que deseja catalogar.

Registrando o Wrapper DB2

Você deve registrar um wrapper para acessar as origens de dados da família DB2. O servidor federado usa o wrapper para se comunicar com as origens de dados e recuperar dados delas. Um wrapper é implementado como um conjunto de arquivos de biblioteca.

Procedimento

Use um dos seguintes métodos para registrar um wrapper:

Método	Descrição
Use o Administration Explorer no Data Studio.	Na pasta Objetos de Banco de Dados Federados , clique com o botão direito do mouse na pasta Wrappers e selecione Criar Wrapper .
Emita a instrução CREATE WRAPPER e especifique o nome para o wrapper.	<p>O nome do wrapper padrão para origens de dados da família DB2 é DRDA. Por exemplo:</p> <pre>CREATE WRAPPER DRDA</pre> <p>Ao usar o nome padrão para registrar o wrapper, não é necessário especificar o nome da biblioteca porque o servidor federado automaticamente usa o nome da biblioteca padrão que está associado ao wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, é possível substituir o nome do wrapper padrão por um nome de sua escolha. Todavia, se você fizer isso, inclua o parâmetro LIBRARY na instrução CREATE WRAPPER.</p> <p>Por exemplo, para registrar um wrapper com o nome db2_wrapper em um servidor federado que use o sistema operacional AIX, emita essa instrução:</p> <pre>CREATE WRAPPER db2_wrapper LIBRARY 'libdb2drda.a'</pre>

O nome da biblioteca padrão é específico para o sistema operacional do servidor federado. Para obter informações adicionais, consulte a lista de arquivos da biblioteca do wrapper DB2.

Arquivos da Biblioteca do Wrapper DB2

Quando você instala o servidor federado, os arquivos da biblioteca do wrapper são incluídos no caminho do diretório padrão.

Se não utilizar o nome padrão do wrapper ao registrá-lo, você deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER e especificar o nome do arquivo da biblioteca padrão do wrapper.

Essa tabela lista os caminhos do diretório padrão e os nomes do arquivo da biblioteca do wrapper padrão. Na tabela *install_path* no caminho do diretório em que o servidor de federação está instalado UNIX ou Linux, e *%DB2PATH%* é a variável de ambiente que especifica o caminho do diretório em que o servidor de federação está instalado no Microsoft Windows. O caminho do diretório padrão do Windows é C:\Program Files\IBM\sqliib.

Tabela 17. Caminhos de Diretório e Nomes dos Arquivos de Biblioteca do DB2 por Sistema Operacional

Sistema Operacional	Caminho do Diretório	Nome do Arquivo de Biblioteca
AIX	/usr/opt/install_path/lib32/ /usr/opt/install_path/lib64/	libdb2drda.a
HP-UX	/opt/IBM/db2/install_path/lib32 /opt/IBM/db2/install_path/lib64	libdb2drda.so
Linux	/opt/IBM/db2/install_path/lib32 /opt/IBM/db2/install_path/lib64	libdb2drda.so
Solaris	/opt/IBM/db2/install_path/lib32 /opt/IBM/db2/install_path/lib64	libdb2drda.so
Windows	%DB2PATH%\bin	db2drda.dll

Para cada sistema operacional, três arquivos de biblioteca são instalados: um arquivo de biblioteca padrão e dois arquivos adicionais que são usados apenas com opções específicas do wrapper.

Registrando as Definições do Servidor para Origens de Dados DB2

O servidor federado necessita de informações de autorização e de senha para se conectar a cada servidor DB2. Como essas informações de autorização e de senha não são armazenadas no catálogo global, você deve incluí-las na definição de cada servidor.

Procedimento

Para registrar uma definição de servidor, use um desses métodos:

Método	Descrição
Use o Administration Explorer no Data Studio.	Na pasta Objetos de Banco de Dados Federados , clique com o botão direito do mouse na pasta Servidores Remotos e selecione Criar Servidor Remoto .

Método	Descrição
Emita a instrução CREATE SERVER	<p>Ao registrar o servidor, você deve incluir determinadas opções de servidor necessárias. Esse exemplo inclui apenas as opções de servidor que são necessárias para registrar um servidor DB2:</p> <pre>CREATE SERVER <i>server_definition_name</i> TYPE <i>server_type</i> VERSION <i>version_number</i> WRAPPER DRDA AUTHORIZATION "<i>userid</i>" PASSWORD "<i>password</i>" OPTIONS (DBNAME '<i>database_name</i>')</pre> <p>DBNAME é uma opção de servidor necessária. O valor de DBNAME é o alias para o banco de dados DB2 que você deseja acessar. Você define o alias ao catalogar o banco de dados.</p>

Nota: Para VERSION, se você tiver usado o DB2 para z/OS Versão 8 para criar o banco de dados em modo de compatibilidade, especifique a Versão 7. Ao registrar o servidor, é possível especificar opções de servidor adicionais na instrução CREATE SERVER. Essas opções incluem opções gerais do servidor e opções do servidor que são específicas para a origem de dados. No InfoSphere Federation Server Versão 9.7 com Fix Pack 2 e posteriores, ao executar a instrução CREATE SERVER, as seguintes opções de servidor são automaticamente configuradas com base na configuração de sua origem de dados:

- DATE_COMPAT
- NUMBER_COMPAT
- SAME_DECFLT_ROUNDING
- VARCHAR2_COMPAT

Se tentar configurar manualmente essas opções do servidor, você receberá a mensagem SQL1841N.

Para obter informações adicionais sobre as opções que podem ser especificadas na instrução CREATE SERVER, consulte as informações de referência das opções do DB2.

O que Fazer Depois

Depois de registrar o servidor, use a instrução ALTER SERVER para incluir opções adicionais do servidor ou descartar opções de servidor existentes.

Instrução CREATE SERVER - Exemplos para o Wrapper DB2

Use a instrução CREATE SERVER para registrar definições do servidor DB2. Esse tópico inclui um exemplo completo com as opções necessárias e um exemplo que mostra o uso de opções adicionais do servidor.

Exemplo Completo

O exemplo a seguir mostra como registrar uma definição de servidor para um wrapper DRDA usando a instrução CREATE SERVER:

```
CREATE SERVER DB2SERVER TYPE DB2/ZOS VERSION 7 WRAPPER DRDA
  AUTHORIZATION "spalten" PASSWORD "db2guru"
  OPTIONS (DBNAME 'CLNTS390')
```

DB2SERVER

Um nome que você designa ao servidor de banco de dados DB2. Nomes de definição de servidor duplicados não são permitidos. Essa é uma opção de servidor necessária.

TYPE DB2/ZOS

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso.

VERSION 7

A versão do servidor de banco de dados DB2 que você deseja acessar.

Nota: Se tiver usado o DB2 para z/OS Versão 8 para criar o banco de dados em modo de compatibilidade, você deve especificar a Versão 7.

WRAPPER DRDA

O nome que você especificou na instrução CREATE WRAPPER.

AUTHORIZATION "spalten"

O ID de autorização na origem de dados. Esse ID deve ter a autoridade BINDADD na origem de dados. Este valor faz distinção entre maiúsculas e minúsculas.

PASSWORD "db2guru"

A senha que está associada ao ID de autorização na origem de dados. Este valor faz distinção entre maiúsculas e minúsculas.

DBNAME 'CLNTS390'

O alias para o banco de dados DB2 que deseja acessar. Esse alias foi definido ao catalogar o banco de dados usando o comando **CATALOG DATABASE**. Este valor faz distinção entre maiúsculas e minúsculas.

Apesar de a variável de nome do banco de dados ser especificada como uma opção na instrução CREATE SERVER, ela é necessária para origens de dados DB2.

Exemplo de Opção do Servidor

Quando você registra a definição do servidor, é possível especificar opções adicionais do servidor na instrução CREATE SERVER. Essas opções incluem opções gerais do servidor e opções de servidor específicas da origem de dados do DB2. Para obter informações adicionais, consulte as informações de referência das opções.

A opção CPU_RATIO indica quão mais rápido ou mais lento a CPU da origem de dados executa em relação à CPU federada. Se você configurar a opção CPU_RATIO como '0.001', isso indica que a CPU na origem de dados remota possui 1.000 vezes mais capacidade disponível que a CPU do servidor federado.

Por exemplo:

```
CREATE SERVER DB2SERVER TYPE DB2/CS VERSION 9.7 WRAPPER DRDA
AUTHORIZATION "spalten" PASSWORD "db2guru"
OPTIONS (DBNAME 'CLNTS390', CPU_RATIO '0.001')
```

Criando Mapeamentos de Usuário para Origens de Dados do DB2

Um mapeamento de usuário define uma associação entre um ID de usuário e uma senha no servidor federado e o ID de usuário e a senha correspondentes no servidor da origem de dados.

Sobre Esta Tarefa

Se os mapeamentos de usuário são necessários ou não para origens de dados do DB2 depende da configuração do ambiente federado. Se o ambiente usar contextos confiáveis federados e autenticação de proxy, nenhum ou apenas poucos mapeamentos de usuário podem ser necessários. Para os melhores resultados, planeje e configure os contextos confiáveis federados antes de criar mapeamentos de usuário.

Procedimento

Use um dos seguintes métodos para criar mapeamentos de usuários para uma origem de dados Oracle:

Método	Descrição
Use o Administration Explorer no Data Studio.	Na pasta Objetos de Banco de Dados Federados , clique com o botão direito do mouse na pasta Mapeamentos de Usuários e selecione Criar Mapeamento de Usuário .
Emita a instrução CRIAR MAPEAMENTO DE USUÁRIO para mapear o ID de usuário local no ID de usuário e senha do servidor do DB2.	Por exemplo: <pre>CREATE USER MAPPING FOR local_userID SERVER server_definition_name OPTIONS (REMOTE_AUTHID 'remote_userID', REMOTE_P 'remote_password')</pre> O REMOTE_AUTHID é o ID de autorização de conexão, não o ID de autorização de ligação.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper DB2

Use a instrução CREATE USER MAPPING para mapear um ID de autorização de servidor em um ID de usuário e uma senha do servidor DB2.

Exemplo Completo

O exemplo a seguir mostra como mapear um ID de autorização de servidor federado em um ID de usuário e uma senha do DB2:

```
CREATE USER MAPPING FOR ALONZO SERVER DB2SERVER  
OPTIONS (REMOTE_AUTHID 'al', REMOTE_PASSWORD 'day2night')
```

ALONZO

Mapeia o ID de autorização local no ID de usuário e na senha remotos.

SERVER DB2SERVER

Especifica o nome do servidor da origem de dados da família DB2 que você definiu na instrução CREATE SERVER.

REMOTE_AUTHID 'al'

Especifica o ID do usuário de conexão no servidor da origem de dados da família DB2 no qual você está mapeando ALONZO. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_ID para 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD 'day2night'

Especifica a senha que está associada com 'al'. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

Exemplo de Registro Especial

A seguir há um exemplo da instrução CREATE USER MAPPING que inclui o registro especial USER:

```
CREATE USER MAPPING FOR USER SERVER DB2SERVER
    OPTIONS (REMOTE_AUTHID 'al', REMOTE_PASSWORD 'day2night')
```

É possível usar o registro especial USER do DB2 para mapear o ID de autorização da pessoa que emite a instrução CREATE USER MAPPING para o ID de usuário da origem de dados especificado na opção de usuário REMOTE_AUTHID.

Exemplo de Mapeamento de Usuário Público

No InfoSphere Federation Server Versão 9.7 Fix Pack 2 e posteriores, é possível criar um mapeamento de usuário público para permitir que todos os usuários do banco de dados local acessem uma origem de dados por meio de um único ID de usuário e uma única senha remotos.

Nesse exemplo, um wrapper DRDA e o servidor de origem de dados do DB2 para z/OS SERVER390 são criados. Em seguida, a instrução USER MAPPING é especificada com PUBLIC para criar um mapeamento de usuário público para o servidor SERVER390. Para mapear todos os usuários do banco de dados local no ID de autorização do servidor SERVER390, o ID de autorização especificado na instrução CREATE SERVER também é especificado em OPTIONS da instrução CREATE USER MAPPING.

Exemplo

```
CREATE WRAPPER DRDA;

CREATE SERVER SERVER390
    TYPE DB2/ZOS VERSION 7.1 WRAPPER DRDA
    AUTHORIZATION "APP_USER" PASSWORD "secret"
    OPTIONS (DBNAME 'remotedb');

CREATE USER MAPPING FOR PUBLIC SERVER SERVER390
    OPTIONS (REMOTE_AUTHID 'APP_USER', REMOTE_PASSWORD 'secret');
```

Testando a Conexão com o Servidor da Origem de Dados do DB2

Teste a conexão com o servidor de origem de dados do DB2 para determinar se o servidor federado está adequadamente configurado para acessar o servidor de origem de dados do DB2.

Procedimento

Para testar a conexão com o servidor DB2, abra uma sessão de passagem e emita uma instrução SQL SELECT nas tabelas de sistema do DB2.

Origem de dados do DB2	Exemplo
DB2 para z/OS	SET PASSTHRU <i>server_definition_name</i> SELECT count(*) FROM sysibm.systables SET PASSTHRU RESET
DB2 para Linux, UNIX e Windows	SET PASSTHRU <i>server_definition_name</i> SELECT count(*) FROM syscat.systables SET PASSTHRU RESET

Origem de dados do DB2	Exemplo
DB2 para System i	<pre>SET PASSTHRU server_definition_name SELECT count(*) FROM qsys2.systables SET PASSTHRU RESET</pre>

Se a instrução SQL SELECT retornar uma contagem, o acesso à origem de dados está configurado adequadamente.

Resolução de Problemas de Erros de Conexão da Origem de Dados

Uma conexão de teste com o servidor de origem de dados pode retornar um erro por diversas razões. Há ações que você pode executar para determinar a causa de um erro.

Sintoma

É retornado um erro quando você tentar se conectar à origem de dados.

Causa

Há diversas causas possíveis para um problema de conexão.

Solução do Problema

Para resolver problemas de erros de conexão com a origem de dados, verifique os seguintes itens:

- Verifique se a origem de dados está disponível.
- Se aplicável, certifique-se de que o servidor de origem de dados esteja configurado para conexões de chegada.
- Certifique-se de suas configurações de mapeamento de usuário para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com a origem de dados. Altere o mapeamento de usuário ou crie outro, conforme necessário.
- Se aplicável, certifique-se de que o software do cliente da origem de dados no servidor federado esteja instalado e configurado corretamente para conexão com a origem de dados.
- Para origens de dados ODBC, certifique-se de que o driver ODBC no servidor federado esteja instalado e configurado corretamente para conexão com o servidor de origem de dados ODBC. Em servidores federados que executam Windows, utilize a ferramenta Administrador de Origem de Dados ODBC para verificar o driver. Em servidores de dados que executam UNIX, consulte a documentação do fornecedor do cliente ODBC.
- Verifique se as configurações para as variáveis configuradas no servidor federado estão corretas para a origem de dados. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis no arquivo db2dj.ini e as variáveis do DB2 Profile Registry (db2set).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e a crie novamente.

Registrando Apelidos para Tabelas e Visualizações do DB2

Para cada definição de servidor do DB2, registre um apelido para cada apelido e visualização que desejar acessar. Em seguida, use os apelidos, não os nomes dos objetos de origem de dados, ao consultar o banco de dados DB2.

Antes de Iniciar

Antes de registrar um apelido, use o comando **RUNSTATS** do DB2 para atualizar as estatísticas na origem de dados do DB2. O servidor federado usa as estatísticas da origem de dados para otimizar o processamento de consulta.

Sobre Esta Tarefa

Para criar um apelido, especifique um nome exclusivo e o nome de seu objeto remoto como, por exemplo, uma tabela ou visualização. Para o DB2 Database para Linux, UNIX e Windows, é possível especificar também o alias de uma tabela, uma visualização ou um apelido. Para o DB2 para z/OS ou DB2 para i/OS, é possível especificar o alias de uma tabela ou visualização.

Restrições

Não é possível criar um apelido em um alias do banco de dados do DB2.

Procedimento

Para registrar um apelido, use um dos seguintes métodos:

Método	Descrição
Use o Administration Explorer no Data Studio.	Na pasta Objetos de Banco de Dados Federados , clique com o botão direito do mouse na pasta Apelidos e selecione Criar Apelido .
Emita a instrução CREATE NICKNAME .	Por exemplo: <code>CREATE NICKNAME <i>nickname</i> FOR <i>server_definition_name</i>."<i>remote_schema</i>."<i>remote.table</i>" ;</code>

Ao criar o apelido, o servidor federado consulta o catálogo de origem de dados utilizando o apelido. Esta consulta testa a conexão com a tabela ou visualização da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

O que Fazer Depois

Depois de concluir esta tarefa, será possível usar apelidos para acessar objetos de origem de dados e executar muitas outras operações, como a criação de um apelido ao longo de uma tabela temporal.

Instrução CREATE NICKNAME - Exemplos para Origens de Dados DB2

Use a instrução **CREATE NICKNAME** e as opções de apelidos necessárias para registrar um apelido para uma tabela ou visualização do DB2 que deseja acessar.

Esse exemplo mostra o uso das opções necessárias para a instrução **CREATE NICKNAME**. Também é possível incluir opções adicionais de apelido e coluna.

```
CREATE NICKNAME DB2SALES FOR DB2SERVER.VINNIE.EUROPE
```

DB2SALES

Um apelido exclusivo que identifica a tabela ou visualização do DB2. O apelido pode incluir tanto um esquema quanto o apelido. Se você omitir o esquema, o ID de autorização do usuário que registra o apelido é usado.

```
DB2SERVER.VINNIE.EUROPE
```

Um identificador de três partes para o objeto remoto:

- *DB2SERVER* é o nome que você designou ao servidor de banco de dados do DB2 na instrução `CREATE SERVER`.
- *VINNIE* é o ID de usuário do proprietário da tabela ou visualização. Este valor faz distinção entre maiúsculas e minúsculas.
- *EUROPE* é o nome da tabela ou visualização remota que você deseja acessar.

Configurando o Acesso às Origens de Dados do Excel

Você pode integrar os dados de origens de dados do Excel às informações de outras origens de dados usando um sistema federado.

Sobre Esta Tarefa

Para configurar um servidor federado para acessar origens de dados do Excel você deve fornecer ao servidor federado informações sobre as origens de dados e os objetos que deseja acessar. Após configurar o servidor federado, você poderá criar consultas para acessar as origens de dados do Excel.

Wrapper do Excel

Uma pasta de trabalho do Excel é um arquivo criado utilizando o aplicativo Microsoft Excel e que possui uma extensão de arquivo xls. O wrapper do Excel é usado para realizar pesquisas nos arquivos do Excel.

Os arquivos do Excel são usados para armazenar informações que são melhor exibidas em uma tabela, com linhas e colunas correspondentes. As pastas de trabalho do Excel consistem em uma ou mais páginas *planilhas*. As planilhas normalmente são usadas para realizar cálculos.

A figura a seguir mostra como o wrapper do Excel conecta suas planilhas ao sistema federado.

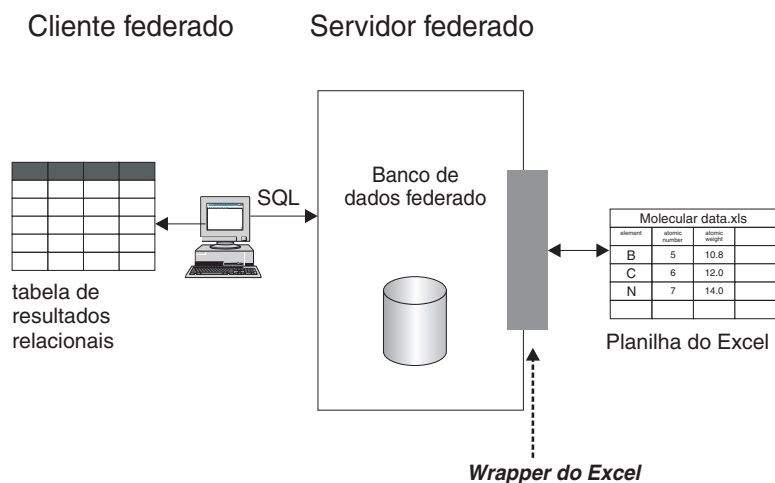


Figura 6. Como Funciona o Wrapper do Excel

O wrapper do Excel usa a instrução `CREATE NICKNAME` para mapear as colunas das planilhas do Excel para colunas do sistema federado. A tabela a seguir contém uma amostra de dados de planilha armazenados em um arquivo denominado `Compound_Master.xls`.

Tabela 18. Amostra de Tabela do Arquivo Compound_Master.xls

	A	B	C	D
1	COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
2	compound_A	1,23	367	tested
3	compound_G		210	
4	compound_F	0,000425536	174	tested
5	compound_Y	1,00256		tested
6	compound_Q		1024	
7	compound_B	33,5362		
8	compound_S	0,96723	67	tested
9	compound_O	1,2		tested

As informações de uma planilha do Excel geralmente não estão disponíveis através de comandos SQL padrão. Quando o wrapper do Excel está instalado e registrado em seu servidor federado, você pode acessar essas informações como se fossem uma fonte de dados relacionais comum. Por exemplo, se você quiser saber todos os dados compostos nos quais a contagem molecular é maior que 100, você deve executar a seguinte consulta SQL:

```
SELECT * FROM compound_master WHERE contagem_mol > 100
```

Os resultados da consulta são mostrados na tabela a seguir.

Tabela 19. Resultados da consulta

COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
compound_A	1,23	367	tested
compound_G		210	
compound_F	0,000425536	174	tested
compound_Q		1024	

Métodos para Acessar Dados do Excel

É possível acessar dados em planilhas do Microsoft Excel utilizando o wrapper do Excel ou o wrapper da ODBC.

Para consultar dados do Excel, ambos os wrappers requerem um servidor federado que possa abrir e ler as planilhas no workbook do Excel. Portanto, o workbook do Excel deve estar no mesmo computador que o servidor federado ou em um drive acessível por rede.

Se você utilizar o wrapper do Excel, o aplicativo do Excel deve ser instalado no servidor federado.

Se você utilizar o wrapper do ODBC, o ODBC driver do Excel deve estar no servidor federado. Este driver é instalado automaticamente com o Microsoft Windows®. O aplicativo do Excel não precisa ser instalado no servidor federado.

Cada wrapper impõe alguns requisitos sobre o local e o layout dos dados nos workbooks do Excel. Com o wrapper do Excel, somente os dados da primeira planilha do workbook podem ser acessados. Com o wrapper do ODBC, você pode acessar dados a partir de qualquer planilha do workbook.

Os exemplos a seguir mostram os requisitos do layout da planilha para esses dois wrappers.

Exemplo de uma Planilha que Contém Linhas de Rótulo e uma Fórmula

Este exemplo mostra uma planilha que contém várias linhas de rótulo no topo da planilha, linhas em branco e uma fórmula na linha 13. Para acessar os dados da planilha, você deve identificar o intervalo de células que deseja acessar.

	A	B	C	D
1	Compound Analysis			
2				
3	Compound Name	Weight	Molecular Count	Tested?
4	compound_A	1.23	367	tested
5	compound_G		210	
6	compound_F	0.000425536	174	tested
7	compound_Y	1.000256		tested
8	compound_Q		1024	
9	compound_B	33.5362		
10	compound_S	0.96723	67	tested
11	compound_O	1.2		tested
12				
13		Total Compounds Tested		5

Figura 7. Uma Planilha que Contém Várias Linhas de Rótulos e uma Fórmula

Se você utilizar o wrapper do Excel

Você especifica o intervalo de células na instrução CREATE NICKNAME utilizando a opção RANGE. Inclua somente os dados no intervalo que você especificar. Não inclua nenhum rótulo da coluna no intervalo. Células que contêm fórmulas, como SUM, retornam o resultado da fórmula e não a fórmula. A menos que queira que os resultados da fórmula sejam retornados, não inclua as células que contêm fórmulas no intervalo. Neste exemplo, o intervalo de células que você inclui na opção RANGE é A4:D11.

Se você utilizar o wrapper do ODBC

Você deve criar um nome para o intervalo de células para designar explicitamente o local dos dados dentro da planilha. O excel se refere a este intervalo de células como um *intervalo nomeado*. O ODBC driver do Excel reconhece somente uma linha de rótulos, a primeira linha do intervalo. Nenhuma linha em branco é permitida entre os rótulos e os dados. O intervalo nomeado deve incluir somente uma linha de rótulos da coluna. Você especifica o intervalo nomeado na instrução CREATE NICKNAME. Você deve incluir uma linha de rótulos da coluna no intervalo que você nomear. Se você não incluir uma linha de rótulos da

coluna no intervalo nomeado, a primeira linha de dados é tratada como rótulos da coluna. Células que contêm fórmulas, como SUM, retornam o resultado da fórmula e não a fórmula. A menos que queira que os resultados da fórmula sejam retornados, não inclua as células que contêm fórmulas no intervalo. Neste exemplo, o intervalo de células que você nomeia é A3:D11.

Exemplo de uma Planilha que Contém uma Linha de Rótulos

Este exemplo mostra uma planilha que contém somente uma linha de rótulos da coluna no topo da planilha. O layout não inclui linhas extras com rótulos, linhas em branco ou células com fórmulas.

	A	B	C	D
1	Compound Name	Weight	Molecular Count	Tested?
2	compound_A	1.23	367	tested
3	compound_G		210	
4	compound_F	0.000425536	174	tested
5	compound_Y	1.000256		tested
6	compound_Q		1024	
7	compound_B	33.5362		
8	compound_S	0.96723	67	tested
9	compound_O	1.2		tested
10				
11				

Figura 8. Uma Planilha que Contém uma Linha de Rótulos da Coluna na Linha 1

Se você utilizar o wrapper do Excel

Você deve especificar o intervalo de células na instrução CREATE NICKNAME utilizando a opção RANGE. O intervalo não pode incluir os rótulos da coluna na linha 1. O intervalo de células que você especificaria é A2:D9.

Se você utilizar o wrapper do ODBC

Você pode acessar esses dados sem criar um intervalo nomeado. Você especifica o nome da planilha na instrução CREATE NICKNAME. O wrapper lê a primeira linha que não esteja em branco conforme rotula e utiliza as informações como nomes de coluna para o pseudônimo. Linhas subsequentes são lidas como dados.

Exemplo de uma Planilha que Contém Somente Dados

Este exemplo mostra uma planilha que contém somente dados. Não existem linhas de rótulos da coluna, linhas em branco ou células com fórmulas.

	A	B	C	D
1	compound_A	1.23	367	tested
2	compound_G		210	
3	compound_F	0.000425536	174	tested
4	compound_Y	1.000256		tested
5	compound_Q		1024	
6	compound_B	33.5362		
7	compound_S	0.96723	67	tested
8	compound_O	1.2		tested
9				
10				

Figura 9. Uma Planilha que Contém Somente Dados

Se você utilizar o wrapper do Excel

Se os dados da primeira planilha do workbook, o wrapper acessará os dados sem utilizar a opção RANGE. Se os dados estiverem em uma outra planilha do workbook, você deve especificar a opção RANGE na instrução CREATE NICKNAME.

Se você utilizar o wrapper do ODBC

Quando você utiliza o wrapper do ODBC para acessar dados do Excel, o wrapper é limitado pelo que o ODBC driver do Excel suporta. O ODBC driver do Excel requer um formato específico para a planilha. O driver supõe que a primeira linha que não está em branco contém os rótulos da coluna. Se a primeira linha que não está em branco contiver dados, os dados de tal linha são tratados como rótulos da coluna para os dados restantes. Se a planilha não contiver uma linha de rótulos da coluna, a primeira linha é utilizada como rótulo e não como dado. Na prática, você perde a primeira linha de dados. Você pode superar esse requisito modificando a planilha. Insira uma nova linha antes dos dados e inclua rótulos para cada coluna de dados para que pareça com o exemplo que contém uma linha de rótulo.

Incluindo Origens de Dados do Excel em um Servidor Federado

Para configurar um servidor federado para acessar origens de dados do Excel você deve fornecer ao servidor federado informações sobre as origens de dados e os objetos que deseja acessar.

Antes de Iniciar

- Os dados de planilhas do Excel devem ser estruturados corretamente para que o wrapper do Excel possa acessá-los.
- A federação deve ser instalada em um servidor que atuará como o servidor federado.
- Um banco de dados deve existir no servidor federado.

Sobre Esta Tarefa

Restrições

- O wrapper do Excel está disponível apenas para as versões do sistema operacional Microsoft Windows que sejam suportadas pelo servidor federado.
- O aplicativo do Excel deve estar instalado no servidor federado.
- A pasta de trabalho do Excel deve estar no mesmo computador que o servidor federado ou em uma unidade acessível pela rede.
- Apenas os dados da primeira planilha de uma pasta de trabalho do Excel podem ser acessados pelo wrapper do Excel.
- O conjunto de páginas de códigos do banco de dados federado deve corresponder ao conjunto de caracteres do arquivo do Excel, caso contrário você poderá obter resultados inesperados de suas consultas.
- Não são permitidas sessões de passagem.

Procedimento

1. “Registrando o Wrapper do Excel”.
2. “Registrando a Definição do Servidor para uma Origem de Dados do Excel” na página 77.
3. “Registrando Apelidos para Origens de Dados do Excel” na página 77.

Registrando o Wrapper do Excel

Você deve registrar um wrapper para acessar origens de dados do Excel.

Sobre Esta Tarefa

Servidores federados utilizam wrappers para se comunicar com e recuperar dados das origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Procedimento

Emita a instrução CRIAR WRAPPER a partir da linha de comandos.

Por Exemplo:

```
CREATE WRAPPER excel_wrapper  
LIBRARY 'db21sxls.dll';
```

Você deve especificar o parâmetro LIBRARY na instrução CREATE WRAPPER.

Arquivos de Biblioteca do Wrapper do Excel:

Os arquivos da biblioteca do wrapper do Excel são incluídos no servidor federado quando este for instalado.

Quando você instala o servidor federado, três arquivos de biblioteca são incluídos no caminho do diretório padrão. Por exemplo, se o servidor federado estiver em execução no Windows, os arquivos da biblioteca do wrapper que são incluídos no caminho de diretório são *db21sxls.dll*, *db21sxlsF.dll* e *db21sxlsU.dll*. O arquivo padrão da biblioteca do wrapper é *db21sxls.dll*. Os outros arquivos de biblioteca do wrapper são utilizados com opções específicas de wrapper.

Quando você registrar o wrapper do Excel, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER e especificar o nome padrão do arquivo de biblioteca do wrapper.

O caminhos padrão do diretório e o nome padrão do arquivo de biblioteca do wrapper estão listados na tabela a seguir.

Tabela 20. Local e Nome do Arquivo de Biblioteca do Wrapper do Excel

Sistema Operacional	Caminho do Diretório	Nome do arquivo da biblioteca do wrapper
Windows	%DB2PATH%\bin	db2lsxls.dll

%DB2PATH% é a variável de ambiente utilizada para especificar o caminho do diretório em que está instalado o servidor federado no Windows. O caminho padrão do diretório do Windows é C:\Arquivos de Programa\IBM\SQLLIB.

Registrando a Definição do Servidor para uma Origem de Dados do Excel

Você deve registrar uma definição de servidor porque a hierarquia dos objetos federados exige que os arquivos de pasta de trabalho do Excel, que são identificados por apelidos, estejam associados a um objeto específico de definição de servidor.

Procedimento

Emita a instrução CRIAR SERVIDOR a partir da linha de comandos.

Por Exemplo:

```
CREATE SERVER server_definition_name  
WRAPPER excel_wrapper;
```

Instrução CREATE SERVER - Exemplos para o Wrapper do Excel:

Utilize a instrução CREATE SERVER para registrar definições de servidor para o wrapper do Excel.

O exemplo a seguir mostra como registrar uma definição de servidor chamada *biochem_lab* para uma pasta de trabalho que contenha dados bioquímicos. A instrução CREATE SERVER que você deve emitir é:

```
CREATE SERVER biochem_lab WRAPPER excel_wrapper;
```

biochem_lab

Um nome atribuído à definição do servidor do Excel. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER *Excel_wrapper*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

Registrando Apelidos para Origens de Dados do Excel

Para cada definição de servidor do Excel registrada, você deverá registrar um apelido para cada tabela do Excel que deseja acessar. Utilize esses apelidos, ao invés dos nomes das planilhas, ao realizar consultas nas origens de dados do Excel.

Sobre Esta Tarefa

Quando você criar um apelido para uma planilha do Excel, as informações dos dados da planilha serão mapeadas para uma tabela relacional.

Células em branco na planilha são interpretadas como o valor NULL.

Podem existir até 10 linhas em branco consecutivas na planilha e elas serão incluídas no conjunto de dados. Mais de 10 linhas em branco consecutivas são interpretadas como o final do conjunto de dados.

Podem existir colunas em branco na planilha. No entanto, essas colunas devem ser registradas e descritas como campos válidos, mesmo que não sejam usadas.

Procedimento

Emita a instrução CRIAR APELIDO a partir da linha de comandos.

Por Exemplo:

```
CREATE NICKNAME nickname
(
  column_name data_type
    OPTIONS (nickname_column_options),
  column_name data_type
    OPTIONS (nickname_column_options),
  column_name data_type
    OPTIONS (nickname_column_options)
)
FOR SERVER server_definition_name
OPTIONS (nickname_options);
```

Os apelidos podem ter até 128 caracteres de comprimento.

O que Fazer Depois

Repita esta etapa para cada tabela do Excel para a qual deseja criar um apelido.

Instrução CREATE NICKNAME - Exemplos para o Wrapper do Excel:

Utilize a instrução CREATE NICKNAME para registrar um apelido para uma tabela do Excel que deseja acessar. Esses exemplos mostram os parâmetros necessários e opções de apelido opcionais.

```
CREATE NICKNAME Compounds
(
  Compound_ID INTEGER,
  CompoundName VARCHAR(50),
  MolWeight FLOAT
)
FOR SERVER biochem_lab
OPTIONS (FILE_PATH 'C:\My Documents\CompoundMaster.xls',
        RANGE 'B2:D25');
```

Compounds

Um apelido exclusivo que é utilizado para identificar a planilha do Excel.

Importante: O apelido é um nome de duas partes que consiste no esquema e no nome do apelido. Se você omitir o esquema quando registrar o apelido, o ID de autorização do usuário que registra o apelido é utilizado para o esquema do apelido.

Compound_ID INTEGER

O nome e o tipo de dados de uma coluna da planilha que contém os identificadores do composto.

CompoundNAME VARCHAR(50)

O nome e o tipo de dados de uma coluna da planilha que contém os nomes dos compostos.

MolWeight *FLOAT*

O nome e o tipo de dados de uma coluna da planilha que contém o peso molecular dos compostos.

FOR SERVER *biochem_lab*

O nome que você atribui à definição do servidor do Excel na instrução CREATE SERVER.

FILE_PATH 'C:\My Documents\CompoundMaster.xls')

Especifica o caminho de diretório e o nome do arquivo completos da pasta de trabalho do Excel que contém os dados que deseja acessar. Os dados devem estar na primeira planilha da pasta de trabalho.

OPTIONS (RANGE 'B2:D25')

Especifica o intervalo de células que você deseja acessar na pasta de trabalho especificada na opção FILE_PATH do apelido.

Qualquer erro de sintaxe ou semântica no valor da opção de intervalo resulta em uma mensagem SQL1882E. Os erros podem incluir:

- O intervalo não é válido. Por exemplo, se a célula superior esquerda especificada no intervalo estiver abaixo ou à direita da célula inferior direita.
- O número de colunas designado pelo intervalo não corresponde ao número de colunas especificado na instrução CREATE NICKNAME.
- Foi localizado um caractere inválido ou outro erro de sintaxe.

Por exemplo, os seguintes caracteres são inválidos:

- #DIV/0!
- #N/A
- #NAME?
- #NULL!
- #NUM!
- #REF!
- #VALUE!

Origens de Dados do Excel - Exemplo de Consultas

Para acessar os dados do Excel, use as colunas de apelido e de apelido definido em suas instruções SQL da mesma forma que usaria nome de tabela e colunas de tabela normais.

Esses exemplos mostram como estruturar as consultas para acessar dados do Excel usando o apelido compounds.

Selecionando uma Coluna Específica de Informações

A consulta a seguir mostra todos os compound_IDs nos quais o peso molecular é maior que 2000:

```
SELECT compound_ID FROM compounds
WHERE molweight > 2000;
```

Usando uma Condição OR nas Instruções SELECT

A consulta a seguir mostra todos os registros nos quais o nome do composto ou o peso molecular é nulo:

```
SELECT * FROM compounds
WHERE compoundname IS NULL OR molweight IS NULL;
```

Usando Condições LIKE e AND em Instruções SELECT

A consulta a seguir mostra todos os registros nos quais o nome composto contém a cadeia ase e o peso molecular é maior ou igual a 300:

```
SELECT * FROM compounds
WHERE compoundname LIKE '%ase%' AND molweight >= 300;
```

Origem de Dados do Excel - Cenário de Amostra

Este cenário mostra as instruções SQL necessárias para registrar os objetos federados usados para acessar uma planilha do Excel. Estão incluídas neste cenário as diferentes consultas que você pode executar usando o apelido que criar.

Informações sobre a Planilha do Excel

Este cenário inicia com uma planilha que contém informações sobre vários compostos. O nome da pasta de trabalho na qual a planilha está incluída é `Compound_Master.xls` e a pasta de trabalho foi criada no Excel. O caminho completo para a pasta de trabalho é `C:\Data\Compound_Master.xls`.

A primeira planilha da pasta de trabalho contém quatro colunas e nove linhas de dados. As colunas listam os nomes dos compostos, o peso dos compostos, a contagem molecular dos compostos e se o composto foi testado.

O conteúdo da planilha é mostrado na tabela a seguir.

Tabela 21. Amostra de planilha Compound_Master.xls

Â	A	B	C	D
1	COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
2	compound_A	1,23	367	tested
3	compound_G	Â	210	Â
4	compound_F	0,000425536	174	tested
5	compound_Y	1,00256	Â	tested
6	compound_Q	Â	1024	Â
7	compound_B	33,5362	Â	Â
8	compound_S	0,96723	67	tested
9	compound_O	1,2	Â	tested

Registrar os Objetos Federados

Para acessar a planilha usando o wrapper do Excel você deve registrar os objetos no servidor federado:

1. Registrar o wrapper do Excel.

Por exemplo:

```
CREATE WRAPPER Excel LIBRARY 'db2lxs.xls.dll';
```

2. Registrar a definição do servidor:

Por exemplo:

```
CREATE SERVER biochem_lab WRAPPER Excel;
```

3. Registrar um apelido que faça referência à planilha do Excel:

Por exemplo:


```

CREATE NICKNAME Compound_Master
  (compound_name VARCHAR(40),
   weight        FLOAT,
   mol_count     INTEGER,
   was_tested   VARCHAR(20))
FOR SERVER biochem_lab
OPTIONS (FILE_PATH 'C:\Data\Compound_Master.xls');

```

O processo de registro foi concluído. A planilha do Excel agora faz parte do sistema federado e pode ser usada em consultas SQL.

Os exemplos a seguir mostram as consultas SQL e os resultados que são retornados para o apelido *Compound_Master*.

Uma Consulta que Retorna Todos os Dados Correspondentes a uma Condição de Cláusula WHERE Específica

Para retornar todos os dados dos compostos que possuem uma contagem molecular maior que 100, emita esta consulta:

```

SELECT * FROM Compound_Master
WHERE mol_count > 100;

```

Uma Consulta que Retorna Colunas Específicas da Planilha

Para retornar os nomes e a contagem molecular de todos os compostos para os quais ainda não foi determinada a contagem molecular, emita este comando:

```

SELECT compound_name, mol_count FROM Compound_Master
WHERE mol_count IS NULL;

```

Todas as colunas das linhas 2, 3, 4, 6 e 8 são retornadas.

As colunas *compound_name* e *mol_count* das linhas 5, 7 e 10 são retornadas.

Uma Consulta que Conta o Número de Linhas que Corresponde a Condições Específicas de Cláusula WHERE

Para retornar o número de compostos que possuem um peso maior que 1 e que não foram testados, emita este comando:

```

SELECT count(*) FROM Compound_Master
WHERE was_tested IS NULL AND weight > 1

```

É retornada uma contagem de registros igual a 1. O composto da linha 7 corresponde aos critérios da consulta.

Uma Consulta que Retorna Colunas Específicas da Planilha e Inclui uma Instrução de Subseleção

Para retornar os nomes e as contagens moleculares de todos os compostos para os quais a contagem molecular foi determinada e esta é menor que a média de contagem molecular, emita esta consulta:

```

SELECT compound_name, mol_count FROM Compound_Master
WHERE mol_count IS NOT NULL
AND mol_count <
  (SELECT AVG(mol_count) FROM Compound_Master
   WHERE mol_count IS NOT NULL AND was_tested IS NOT NULL);

```

A subconsulta retorna 368 para a média de contagem molecular. A consulta principal usa a média para retornar os resultados da consulta, mostrados na tabela a seguir:

Tabela 22. Resultados da consulta

COMPOUND_NAME	MOL_COUNT
compound_A	367
compound_G	210
compound_F	174
compound_S	67

Modelo de Controle de Acesso do arquivo para o Wrapper do Excel

Para acessar um arquivo do Excel, o wrapper precisa de uma identidade de usuário por motivos de segurança. O wrapper do Excel usa a identidade do usuário associado ao serviço de banco de dados federado. O nome do serviço do banco de dados federado depende do nome da instância do banco de dados. Por exemplo, se o nome da instância do banco de dados for DB2, o nome do serviço será DB2 - DB2. Para determinar a identidade do usuário associado ao serviço do banco de dados federado, use o Pannel de controle do Windows para exibir os serviços. Dê um clique duplo no nome do serviço e abra a página de propriedades de logon.

Configurando o Acesso a Origens de Dados do Informix

Para configurar um servidor federado para acessar origens de dados do Informix você deve fornecer ao servidor federado informações sobre as origens de dados e os objetos que deseja acessar.

Antes de Iniciar

- O software SDK do Informix Client deve ser instalado e configurado no servidor que agirá como servidor federado.
- Verifique a configuração do servidor federado.
- Verifique o parâmetro federado para assegurar que a federação está ativada.
- Em servidores federados AIX, o AIX Base Application Development Math Library deve ser instalado. Você pode determinar se a Biblioteca está instalada emitindo o comando do AIX, `ls1pp -l bos.adt.libm`.

Sobre Esta Tarefa

Restrições

The Informix wrapper is not supported in trusted mode on SuSE Linux Enterprise Server, Version 9.2 or later.

Procedimento

1. Configure e teste o arquivo de configuração do cliente Informix.
2. Defina as variáveis de ambiente do Informix.
3. Registre o wrapper.
4. Registre a definição do servidor.
5. Crie os mapeamentos de usuário.

6. Teste a conexão com o servidor.
7. Registre apelidos para tabelas, visualizações e sinônimos Informix.

Configurando e Testando o Arquivo de Configuração do Cliente Informix

O arquivo de configuração do cliente Informix é utilizado para conectar-se a bancos de dados Informix utilizando as bibliotecas do cliente instaladas no servidor federado.

Antes de Iniciar

O software SDK do Informix Client deve ser instalado no servidor federado.

Sobre Esta Tarefa

O arquivo de configuração do cliente especifica o local de cada servidor de banco de dados Informix e tipo de conexão (protocolo) para o servidor de banco de dados.

O local padrão do arquivo de configuração do cliente depende do sistema operacional utilizado pelo servidor federado.

- Em servidores federados que executam UNIX, o local e nome padrão do arquivo é \$INFORMIXDIR/etc/sqlhosts. O arquivo sqlhosts é instalado com o SDK do Informix Client.
- Em servidores federados que executam Windows, o local padrão do registro sqlhosts é o computador local.

O formato do sqlhosts é descrito no manual *Administrators Guide for Informix Dynamic Server*.

Procedimento

1. Configure o SDK do Informix Client.
 - Em servidores federados que executam UNIX, você pode configurar o SDK do Informix Client editando o arquivo sqlhosts. Você pode também copiar o arquivo sqlhosts de um outro sistema que possua o Informix Connect ou SDK do Informix Client instalado.
 - Em servidores federados que executam Windows, você pode configurar o SDK do Informix Client com o utilitário Setnet32 Informix. O utilitário Setnet32 configura o registro sqlhosts.
2. Verifique o local do arquivo ou registro sqlhosts.
 - Em servidores federados que executam UNIX, o arquivo sqlhosts está localizado no diretório \$INFORMIXDIR/etc/.
 - Em servidores federados que executam Windows, as informações sobre sqlhosts são mantidas na seguinte chave de registro do Windows :
HKEY_LOCAL_MACHINE\SOFTWARE\INFORMIX\SQLHOSTS
3. Se desejar colocar o arquivo ou registro sqlhosts em um caminho que não seja o caminho da procura padrão, defina a variável de ambiente INFORMIXSQLHOSTS para especificar o local do arquivo. Utilize uma das opções a seguir para definir a variável de ambiente INFORMIXSQLHOSTS.
 - Em servidores federados que executam UNIX, defina a variável de ambiente INFORMIXSQLHOSTS para o nome completo do arquivo sqlhosts.

- Em servidores federados que executam Windows, utilize o utilitário Setnet32 para definir a variável de ambiente INFORMIXSQLHOSTS para o nome do computador Windows que armazena o registro.
4. Teste a conexão para assegurar que o software cliente é capaz de conectar-se ao servidor Informix. Se o utilitário dbaccess Informix estiver no servidor federado, utilize essa ferramenta para testar a conexão. Caso contrário, execute o programa demo Informix para testar a configuração do cliente.

Definindo as Variáveis de Ambiente do Informix

As variáveis de ambiente Informix devem ser definidas no arquivo `db2dj.ini` no servidor federado.

Sobre Esta Tarefa

Existem variáveis de ambiente requeridas e opcionais para origens de dados Informix. Se você instalou o software cliente Informix antes de instalar o wrapper do Informix, as variáveis de ambiente requeridas do Informix estarão definidas no arquivo `db2dj.ini`.

Você deve definir as variáveis de ambiente utilizando as etapas nesta tarefa caso não tenha instalado o software cliente Informix antes de instalar o wrapper do Informix ou se desejar definir qualquer das variáveis de ambiente opcionais.

Procedimento

1. Utilize um dos métodos a seguir para definir as variáveis de ambiente do Informix que deseja utilizar:

Método	Etapa
Para definir as variáveis de ambiente automaticamente	Execute o programa de configuração do DB2 novamente e especifique a opção de instalação Customizada . Siga as instruções no assistente. Importante: Executar o programa de instalação novamente definirá somente as variáveis de ambiente requeridas. As variáveis de ambiente opcionais devem ser definidas manualmente.
To set the environment variables with the wrapper configuration tool	Execute a ferramenta de configuração do wrapper para configurar as variáveis de ambiente necessárias e as opcionais. Follow the instructions in "Executando a Ferramenta de Configuração de Wrapper" na página 1.
Para definir variáveis de ambiente manualmente	Edite o arquivo <code>db2dj.ini</code> . <ul style="list-style-type: none"> • Em servidores federados que executam UNIX, o arquivo <code>db2dj.ini</code> está localizado no diretório <code>sql1lib/cfg</code>. • Em servidores federados que executam Windows, o arquivo <code>db2dj.ini</code> está localizado no diretório <code>%DB2PATH%\cfg</code>.

O arquivo `db2dj.ini` contém informações de configuração sobre o software cliente Informix que está instalado no servidor federado. Se o arquivo não existir, você poderá criar um arquivo com o nome `db2dj.ini` utilizando qualquer editor de texto. No arquivo `db2dj.ini`, você deverá especificar o

caminho completo para as variáveis de ambiente; caso contrário encontrará erros. As seguintes variáveis de ambiente mostram como uma entrada no arquivo `db2dj.ini` no UNIX pode ser:

```
INFORMIXDIR=/informix/csdk  
INFORMIXSERVER=inf10
```

2. Defina as variáveis de ambiente de conversão de página de códigos do Informix (se necessário).
3. Para assegurar que as variáveis de ambiente estejam configuradas no servidor federado, recicle a instância do banco de dados federado.

Emita os seguintes comandos para reciclar a instância do banco de dados federado:

```
db2stop  
db2start
```

Variáveis de Ambiente do Informix

Existem variáveis de ambiente requeridas e opcionais para origens de dados Informix. Essas variáveis são definidas no arquivo `db2dj.ini`.

A variáveis de ambiente válidas para o Informix são:

- INFORMIXDIR
- INFORMIXSERVER
- INFORMIXSQLHOSTS (opcional)
- CLIENT_LOCALE (opcional)
- DB_LOCALE (opcional)
- DBNLS (opcional)

As variáveis de ambiente `CLIENT_LOCALE`, `DB_LOCALE` e `DBNLS` são variáveis de ambiente da página de códigos.

Descrições da Variável

INFORMIXDIR

Especifica o caminho do diretório em que o software SDK do Informix Client está instalado.

Por Exemplo:

- Em servidores federados que executam UNIX, defina o caminho como:
`INFORMIXDIR=/informix/csdk`
- Em servidores federados que executam Windows, defina o caminho como:
`INFORMIXDIR=C:\informix\csdk`

INFORMIXSERVER

Identifica o nome do servidor Informix padrão. Essa configuração deve ser uma entrada válida no arquivo `sqlhosts` (UNIX) ou a chave de registro `SQLHOSTS` no (Windows). Para obter um valor para `INFORMIXSERVER`, leia o arquivo `sqlhosts`. Selecione um dos valores `dbservername`. O `dbservername` é o primeiro valor em cada entrada no arquivo `sqlhosts`.

Por Exemplo:

```
INFORMIXSERVER=inf10
```

Requisito: Embora o wrapper do Informix não utilize o valor desta variável de ambiente, o cliente Informix requer que esta variável de

ambiente seja definida. O wrapper utiliza o valor da opção do servidor NODE, o qual especifica o servidor de banco de dados Informix que deseja acessar.

INFORMIXSQLHOSTS

Se você estiver utilizando o caminho padrão para o arquivo sqlhosts no Informix, não é necessário configurar essa variável de ambiente. No entanto, se estiver utilizando algum outro caminho para o arquivo sqlhosts no Informix, será necessário definir essa variável de ambiente. Defina a variável INFORMIXSQLHOSTS como o nome do caminho completo em que o arquivo sqlhosts reside no Informix.

- Em servidores federados que executam UNIX, o caminho padrão é \$INFORMIXDIR/etc/sqlhosts.
- Em servidores federados que executam Windows, se a chave de registro SQLHOSTS não estiver no computador local, o valor da variável de ambiente INFORMIXSQLHOSTS é o nome do computador Windows que armazena o registro.

Um exemplo UNIX de definição dessa variável de ambiente como um outro caminho é:

```
INFORMIXSQLHOSTS=/informix/csdk/etc/my_sqlhosts
```

Conversão da Página de Códigos do Informix:

Cada vez que o wrapper do Informix se conecta a uma origem de dados do Informix, o wrapper determina qual valor de página de códigos utilizar para essa conexão. Você pode fazer com que o wrapper do Informix defina o valor da página de códigos ou designar uma página de códigos definindo a variável de ambiente CLIENT_LOCALE.

As variáveis de ambiente que especificam a conversão da página de códigos do Informix estão definidas no arquivo db2dj.ini no servidor federado.

Para conversão da página de códigos do Informix, você pode definir as seguintes variáveis de ambiente opcionais:

- CLIENT_LOCALE
- DB_LOCALE
- DBNLS

As variáveis de ambiente da página de códigos do Informix são:

CLIENT_LOCALE

Especifica o código do idioma do Informix que você deseja utilizar. Utilize essa variável quando não desejar que o wrapper do Informix determine automaticamente a configuração da variável.

Por Exemplo:

```
CLIENT_LOCALE=Informix_client_locale_value
```

- Se a variável CLIENT_LOCALE estiver definida no arquivo db2dj.ini no servidor federado, o wrapper utiliza o valor da página de códigos no arquivo db2dj.ini.
- Se a variável CLIENT_LOCALE não estiver definida no servidor federado, o wrapper determina o território e a página de códigos do banco de dados federado. O wrapper define a variável CLIENT_LOCALE para o código do idioma do Informix mais próximo. Se não houver código do idioma correspondente no Informix o wrapper

define a variável `CLIENT_LOCALE` como o código do idioma `en_us.8859-1` para sistemas UNIX e como o código do idioma `en_us.CP1252` para sistemas Windows.

Você pode visualizar a lista de códigos do idioma válidos Informix emitindo o comando `gfiles` no servidor do Informix s.

Consulte o *Informix Guide to GLS Functionality* para obter mais informações sobre conversões de páginas de códigos.

DB_LOCALE

Especifica que o banco de dados Informix utiliza uma página de códigos diferente do código de idioma do cliente. Utilize essa variável quando desejar que o Informix desempenhe conversões entre as duas páginas de códigos. Defina a variável de ambiente `DB_LOCALE` como o nome do código do idioma do banco de dados Informix.

Por Exemplo:

```
DB_LOCALE=Informix_db_locale_value
```

DBNLS

Especifica se o Informix verifica se a configuração do `DB_LOCALE` corresponde ao código do idioma real do banco de dados do Informix. Defina esta variável de ambiente como 1.

Por Exemplo:

```
DBNLS=1
```

Force o Informix a desempenhar a conversão da página de códigos

O banco de dados do Informix utiliza uma página de códigos diferente do código de idioma do cliente e você deseja que o Informix desempenhe conversões entre as duas páginas. Você precisa:

1. Definir a variável de ambiente `DB_LOCALE` do Informix como o nome do código do idioma do banco de dados do Informix. Você define essa variável no arquivo `db2dj.ini` no servidor federado.
2. Para verificar se a configuração da `DB_LOCALE` corresponde ao código do idioma real do banco de dados do Informix, defina a variável de ambiente `DBNLS` do Informix como 1. Você define essa variável no arquivo `db2dj.ini` no servidor federado.

Os Dados do Informix que Utilizam a Página de Códigos em Chinês GB 18030

Para acessar dados que utilizam a página de códigos em chinês GB 18030, utilize a página de códigos UTF-8 no banco de dados federado e inclua a seguinte configuração no arquivo `db2dj.ini` para que o Informix traduza corretamente os dados da GB 18030 ao unicode.

```
DB_LOCALE=zh_cn.GB18030-2000
```

Registrando o Wrapper do Informix

Você deve registrar um wrapper para acessar as origens de dados do Informix. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Procedimento

Emita a instrução CREATE WRAPPER e especifique o nome padrão para o wrapper do Informix.

Por exemplo:

```
CREATE WRAPPER INFORMIX;
```

Recomendação: Utilize o nome padrão do wrapper. O nome padrão do wrapper para Informix é INFORMIX. Ao registrar um wrapper utilizando o nome padrão, o servidor federado utiliza automaticamente a biblioteca de wrapper apropriada do Informix para o sistema operacional no qual o servidor federado está em execução. Se o nome padrão do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, você poderá substituir o nome padrão do wrapper por um de sua escolha. Se você não utilizar o nome padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.

Por exemplo, para registrar um wrapper com o nome de `informix_wrapper` em um servidor federado que utiliza o sistema operacional AIX, emita a seguinte instrução:

```
CREATE WRAPPER informix_wrapper LIBRARY 'libdb2informix.a';
```

O nome do arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado. Consulte a lista de arquivos de biblioteca do wrapper do Informix para obter o nome de biblioteca correto a ser especificado na instrução CREATE WRAPPER.

Arquivos da Biblioteca do Wrapper do Informix

Os arquivos da biblioteca do wrapper do Informix são incluídos no servidor federado quando você instala federação.

Quando você instala federação, três arquivos de biblioteca são incluídos no caminho do diretório padrão. Por exemplo, se o servidor federado estiver em execução no AIX, os arquivos de biblioteca do wrapper incluídos no caminho do diretório são `libdb2informix.a`, `libdb2informixF.a` e `libdb2informixU.a`. O arquivo de biblioteca do wrapper padrão é `libdb2informix.a`.

Se você decidir não utilizar o nome padrão do wrapper ao registrar um wrapper, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER e especificar o nome do arquivo de biblioteca padrão do wrapper.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 23. Locais de Biblioteca do Wrapper e Nomes de Arquivos do Informix

Sistema Operacional	Caminho do Diretório	Nome do Arquivo de Biblioteca
AIX	<code>/usr/opt/install_path/lib32/</code> <code>/usr/opt/install_path/lib64/</code>	<code>libdb2informix.a</code>
HP-UX	<code>/opt/IBM/db2/install_path/lib32</code> <code>/opt/IBM/db2/install_path/lib64</code>	<code>libdb2informix.so</code>
Linux	<code>/opt/IBM/db2/install_path/lib32</code> <code>/opt/IBM/db2/install_path/lib64</code>	<code>libdb2informix.so</code>
Solaris	<code>/opt/IBM/db2/install_path/lib32</code> <code>/opt/IBM/db2/install_path/lib64</code>	<code>libdb2informix.so</code>

Tabela 23. Locais de Biblioteca do Wrapper e Nomes de Arquivos do Informix (continuação)

Sistema Operacional	Caminho do Diretório	Nome do Arquivo de Biblioteca
Windows	%DB2PATH%\bin	db2informix.dll

- *install_path* é o caminho do diretório onde a federação é instalada no UNIX ou Linux.
- %DB2PATH% é a variável de ambiente que é utilizada para especificar o caminho do diretório onde a federação é instalada no Windows. O caminho padrão do caminho do diretório do Windows é C:\Arquivos de Programa\IBM\SQLLIB.

Registrando as Definições do Servidor para uma Origem de Dados do Informix

Você deverá registrar cada servidor do Informix que deseja acessar no banco de dados federado.

Procedimento

1. Localize o nome do nó no arquivo ou registro sqlhosts do Informix.

Amostra de arquivo sqlhosts :

```
inf10an onsoctcp anaconda inmx10
inf10bo onsoctcp boa ifmx10
inf10py onsoctcp python ifmx10
```

- O primeiro valor em cada linha é o *node_name*, tal como inf10an.
- O segundo valor em cada linha é o *nettype*, ou tipo de conexão. Neste exemplo onsoctcp indica que esta é uma conexão TCP/IP.
- O terceiro valor em cada linha é o nome do host, como anaconda, boa e python.
- O quarto valor em cada linha é o nome do serviço, tal como inmx10. O campo nome do serviço depende do *nettype* listado no segundo valor.

Para obter informações adicionais sobre o formato do arquivo sqlhosts e o significado desses campos, consulte o manual do Informix *Administrators Guide for Informix Dynamic Server*.

2. Emita a instrução CRIAR SERVIDOR a partir da linha de comandos.

Por exemplo:

```
CREATE SERVER server_definition_name TYPE informix
    VERSION version_number WRAPPER INFORMIX
    OPTIONS (NODE 'node_name', DBNAME 'database_name');
```

Embora as variáveis '*node_name*' e '*database_name*' sejam especificadas como opções na instrução CREATE SERVER, essas opções são requeridas para origens de dados Informix.

Após o registro da definição do servidor, utilize a instrução ALTER SERVER para incluir ou descartar opções do servidor.

Instrução CREATE SERVER - Exemplos para o Wrapper do Informix

Utilize a Instrução CREATE SERVER para Registrar Definições de Servidor para o WrapperInformix. Este tópico inclui um exemplo completo com os parâmetros e exemplos com opções adicionais do servidor.

Exemplo Completo

O Exemplo a Seguir Mostra como Registrar uma Definição de Servidor para um Wrapper Informix Utilizando a Instrução CREATE SERVER:

```
CREATE SERVER asia TYPE informix VERSION 10 WRAPPER INFORMIX
      OPTIONS (NODE 'abc', DBNAME 'sales');
```

asia Um nome que você atribui ao servidor de banco de dados Informix. Nomes de definição de servidor duplicados não são permitidos.

TYPE *informix*

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso. Para o wrapper do Informix o tipo de servidor deve ser *informix*.

VERSION 10

A Versão do Servidor de Banco de Dados Informix que Deseja Acessar.

WRAPPER *INFORMIX*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

NODE '*abc*'

O Nome do Nó em que se Encontra o Servidor de Banco de Dados Informix. Obtenha o nome do nó a partir do arquivo `sqlhosts`. Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, é requerida para origens de dados do Informix.

DBNAME '*sales*'

O Nome do Banco de Dados Informix que Deseja Acessar. Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do banco de dados seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dados do Informix.

Opções Adicionais de Servidor

Quando você cria uma definição do servidor, é possível especificar opções adicionais do servidor na instrução CREATE SERVER. As opções do servidor podem ser gerais ou específicas do Informix.

Opções do Servidor FOLD_ID e FOLD_PW

Quando o servidor federado se conecta a uma origem de dados, o servidor federado tenta se conectar utilizando todas as combinações possíveis de maiúsculas e minúsculas para o ID de usuário e a senha, bem como as maiúsculas e minúsculas atuais. O servidor federado pode fazer até nove tentativas de conexão antes da conexão com êxito ao servidor de origem de dados. Tais tentativas podem tornar os tempos de conexão mais lentos e podem resultar no bloqueio do ID de usuário. Você pode evitar bloqueios especificando os valores para as opções do servidor FOLD_ID e FOLD_PW. Você pode definir as opções do servidor FOLD_ID e FOLD_PW como 'N' (não feche o ID e a senha).

Se definir as opções do servidor FOLD_ID e FOLD_PW para 'N', você deve especificar o ID e a senha do usuário com minúsculas e maiúsculas corretas. A vantagem em definir essas opções do servidor como 'N' é que quando um ID de usuário ou senha inválidos forem especificados, o wrapper não continuará

tentando as várias combinações de maiúsculas e minúsculas. Essas duas opções do servidor podem reduzir a chance de exceder o número máximo de tentativas falhas de efetuar login e do ID ser bloqueado.

O exemplo a seguir mostra uma definição de servidor Informix com essas opções de servidor:

```
CREATE SERVER asia TYPE informix VERSION 10 WRAPPER INFORMIX
  OPTIONS (NODE 'abc', DBNAME 'sales', FOLD_ID 'N', FOLD_PW 'N');
```

Exemplo de Opção do Servidor IUD_APP_SVPT_ENFORCE

A opção IUD_APP_SVPT_ENFORCE especifica se o servidor federado deve reforçar a detecção ou construção de instruções de savepoint do aplicativo. O Informix não suporta instruções de savepoint do aplicativo. Quando definido como 'N', o servidor federado não realizará as transações novamente quando for encontrado erro. O aplicativo deve manipular a recuperação de erro.

A opção do servidor IUD_APP_SVPT_ENFORCE deve ser definida como 'N' para ativar a replicação para ou a partir de origens de dados Informix. O exemplo a seguir mostra uma definição de servidor Informix com a opção do servidor IUD_APP_SVPT_ENFORCE.

```
CREATE SERVER asia TYPE informix VERSION 10 WRAPPER INFORMIX
  OPTIONS (NODE 'abc', DBNAME 'sales', IUD_APP_SVPT_ENFORCE 'N');
```

Opções INFORMIX_DB_LOCALE e INFORMIX_CLIENT_LOCALE

A opção INFORMIX_DB_LOCALE define a variável de ambiente do código do idioma do banco de dados (DB_LOCALE) que é utilizada para a conexão entre o servidor federado e o servidor de origem de dados. Se a opção INFORMIX_DB_LOCALE não estiver especificada, a variável de ambiente Informix DB_LOCALE é definida como o valor especificado no arquivo db2dj.ini. Se o arquivo db2dj.ini não especificar a variável de ambiente DB_LOCALE, a variável de ambiente DB_LOCALE do Informix não é definida. Um valor válido é qualquer código do idioma Informix válido. Esta opção é opcional. A definição padrão é Nenhum(a).

A opção INFORMIX_CLIENT_LOCALE define a variável de ambiente do código do idioma do cliente (CLIENT_LOCALE) a ser utilizada para a conexão entre o servidor federado e o servidor de origem de dados. Se a opção INFORMIX_CLIENT_LOCALE não estiver especificada, a variável de ambiente Informix CLIENT_LOCALE é definida como o valor especificado no arquivo db2dj.ini. Se o arquivo db2dj.ini não especificar CLIENT_LOCALE, a variável de ambiente CLIENT_LOCALE do Informix é definida como o código do idioma Informix que mais se aproxima da página de códigos e do território do banco de dados federado. Um valor válido é qualquer código do idioma Informix válido. Esta opção é opcional. A definição padrão é Nenhum(a).

O Exemplo a seguir mostra uma definição de servidor Informix com as opções INFORMIX_DB_LOCALE e INFORMIX_CLIENT_LOCALE.

```
CREATE SERVER asia TYPE informix VERSION 10 WRAPPER INFORMIX
  OPTIONS (NODE 'abc', DBNAME 'sales', INFORMIX_DB_LOCALE 'en_us.8859-1',
  INFORMIX_CLIENT_LOCALE 'en_us.CP1252');
```

Criando os Mapeamentos de Usuário para uma Origem de Dados do Informix

Quando você tenta acessar um servidor Informix, o servidor federado estabelece uma conexão com o servidor Informix utilizando um ID de usuário e uma senha válidos para a origem de dados. Você deve definir uma associação (um mapeamento de usuário) entre o ID de usuário e a senha de cada servidor federado e o ID de usuário e a senha correspondentes da origem de dados.

Sobre Esta Tarefa

Crie um mapeamento de usuário para cada ID de usuário que acessará o sistema federado para enviar pedidos distribuídos à origem de dados do Informix.

Procedimento

Emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR local_userID SERVER server_definition_name
    OPTIONS (REMOTE_AUTHID 'remote_userID', REMOTE_PASSWORD 'remote_password');
```

Embora as variáveis REMOTE_AUTHID e REMOTE_PASSWORD estejam especificadas como opções na instrução CREATE USER MAPPING, tais opções são necessárias para acessar as origens de dados do Informix.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper do Informix

Utilize a instrução CREATE USER MAPPING para mapear um ID de usuário do servidor federado para um ID de usuário e senha do servidor do Informix. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o USER de registro especial do DB2 com a instrução CREATE USER MAPPING.

Exemplo Completo

O exemplo a seguir mostra como mapear um ID de usuário do servidor federado para um ID de usuário e senha do servidor do Informix :

```
CREATE USER MAPPING FOR VINCENT SERVER asia
    OPTIONS (REMOTE_AUTHID 'vinnie', REMOTE_PASSWORD 'close2call');
```

VINCENT

Especifica o ID de usuário local que você está mapeando para um ID de usuário definido no servidor Informix.

SERVER asia

Especifica o nome da definição do servidor registrado na instrução CREATE SERVER para o servidor Informix.

REMOTE_AUTHID 'vinnie'

Especifica o ID de usuário no servidor de banco de dados Informix para o qual você está mapeando *VINCENT*. O valor fará distinção entre maiúsculas e minúsculas, exceto se você configurar a opção do servidor FOLD_ID como 'U' ou 'L' na instrução CREATE SERVER.

Embora o ID de usuário remoto seja especificado como uma opção na instrução CREATE USER, é requerida para origens de dados do Informix.

REMOTE_PASSWORD 'close2call'

Especifica a senha associada a '*vinnie*'. O valor faz distinção entre

maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

Embora a senha remota seja especificada como uma opção na instrução CREATE USER MAPPING, ela é requerida para origens de dados do Informix.

Exemplo de Registro Especial

Você pode utilizar o USER de registro especial do DB2 para mapear o ID de autorização da pessoa que está emitindo a instrução CREATE USER MAPPING para o ID de autorização da origem de dados especificado na opção de usuário REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER SERVER asia
  OPTIONS (REMOTE_AUTHID 'vinnie', REMOTE_PASSWORD 'close2call');
```

Testando a Conexão com o Servidor Informix

Teste a conexão com o servidor de origem de dados do Informix para determinar se o servidor federado está configurado adequadamente para acessar as origens de dados do Informix.

Sobre Esta Tarefa

Você pode testar a conexão com o servidor do Informix utilizando a definição do servidor e os mapeamentos de usuário que definiu.

Procedimento

Abra uma sessão de passagem e emita uma instrução SELECT nas tabelas do sistema Informix. Se a instrução SELECT retornar uma contagem, a definição do servidor e o mapeamento de usuário estão configurados apropriadamente.

Por exemplo:

```
SET PASSTHRU server_definition_name
SELECT count(*) FROM informix.systables
SET PASSTHRU RESET
```

Se a instrução SELECT retornar um erro, você deverá resolver os problemas dos erros de conexão.

Resolução de Problemas de Erros de Conexão da Origem de Dados

Uma conexão de teste com o servidor de origem de dados pode retornar um erro por diversas razões. Há ações que você pode executar para determinar a causa de um erro.

Sintoma

É retornado um erro quando você tentar se conectar à origem de dados.

Causa

Há diversas causas possíveis para um problema de conexão.

Solução do Problema

Para resolver problemas de erros de conexão com a origem de dados, verifique os seguintes itens:

- Verifique se a origem de dados está disponível.
- Se aplicável, certifique-se de que o servidor de origem de dados esteja configurado para conexões de chegada.
- Certifique-se de suas configurações de mapeamento de usuário para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com a origem de dados. Altere o mapeamento de usuário ou crie outro, conforme necessário.
- Se aplicável, certifique-se de que o software do cliente da origem de dados no servidor federado esteja instalado e configurado corretamente para conexão com a origem de dados.
- Para origens de dados ODBC, certifique-se de que o driver ODBC no servidor federado esteja instalado e configurado corretamente para conexão com o servidor de origem de dados ODBC. Em servidores federados que executam Windows, utilize a ferramenta Administrador de Origem de Dados ODBC para verificar o driver. Em servidores de dados que executam UNIX, consulte a documentação do fornecedor do cliente ODBC.
- Verifique se as configurações para as variáveis configuradas no servidor federado estão corretas para a origem de dados. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis no arquivo db2dj.ini e as variáveis do DB2 Profile Registry (db2set).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e a crie novamente.

Ajuste de Desempenho para o Wrapper do Informix

Você pode utilizar as opções do servidor FOLD_ID e FOLD_PW para aprimorar a conectividade entre o servidor federado e as origens de dados do Informix.

Quando o servidor federado se conecta a uma origem de dados, o servidor tenta se conectar utilizando todas as combinações possíveis de maiúsculas e minúsculas para o ID de usuário e a senha. O servidor pode fazer até nove tentativas de conexão antes de conectar com êxito ao servidor de origem de dados. Tais tentativas podem tornar os tempos de conexão mais lentos e podem resultar no bloqueio do ID de usuário.

Você pode aprimorar o desempenho especificando os valores para as opções do servidor FOLD_ID e FOLD_PW.

- Se todos os seus IDs e senhas de usuário do Informix estiverem em minúsculas, a definição das opções do servidor FOLD_ID e FOLD_PW com o valor 'L' pode aprimorar o tempo de conexão.

Por exemplo:

```
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_ID 'L');
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_PW 'L');
```

- O servidor federado tenta cada combinação de valores de maiúsculas e minúsculas para o ID de usuário e a senha. Você pode reduzir a chance do número máximo de tentativas de efetuar login falhas ser excedido definindo essas opções como 'N' (não feche o ID e a senha do usuário). Se você estabelecer essas configurações, precisará sempre especificar o ID de usuário e a senha com

as minúsculas e maiúsculas corretas. Se um ID e senha inválidos forem especificados, o wrapper não continuará tentando as várias combinações.

Por exemplo:

```
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_ID 'N');
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_PW 'N');
```

Registrando Apelidos para Tabelas, Visualizações e Sinônimos do Informix

Para cada definição de servidor do Informix registrada, você deverá registrar um apelido para cada tabela, visualização ou sinônimo que deseja acessar. Utilize esses apelidos em vez dos nomes dos objetos de origem de dados quando consultar os servidores do Informix.

Antes de Iniciar

Atualize as estatísticas na origem de dados do Informix antes de registrar um apelido. O banco de dados federado confia nas estatísticas do catálogo da origem de dados para otimizar o processamento da consulta. Pode ser utilizado o comando **UPDATE STATISTICS** do Informix, que equivale ao comando **RUNSTATS** do DB2, para atualizar as estatísticas da origem de dados.

Procedimento

Emita a instrução **CREATE NICKNAME**. Os apelidos podem ter até 128 bytes de comprimento.

Por exemplo:

```
CREATE NICKNAME nickname FOR server_definition_name."remote_schema"."remote.table" ;
```

Ao criar um apelido, o servidor federado consulta o catálogo de origem de dados utilizando o apelido. Esta consulta testa a conexão com a tabela, visualização ou sinônimo da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

O que Fazer Depois

Repita esta etapa para cada tabela, visualização ou sinônimo do Informix para as quais deseja criar um apelido.

Instrução **CREATE NICKNAME** - Exemplos para o Wrapper do Informix

Utilize a instrução **CREATE NICKNAME** para registrar um apelido para uma tabela visualização ou sinônimo do Informix, que deseja acessar. Este tópico inclui um exemplo completo com os parâmetros requeridos.

Exemplo Completo

```
CREATE NICKNAME JPSALES FOR asia."vinnie"."japan" ;
```

JPSALES

Um apelido exclusivo que é utilizado para identificar a tabela, visualização ou sinônimo do Informix.

Importante: O apelido é um nome de duas partes -o esquema e o apelido. Se você omitir o esquema ao registrar o apelido, o esquema do apelido será o ID de autorização do usuário que registrar o apelido.

asia."vinnie"."japan"

Um identificador de três partes para o objeto remoto:

- *asia* é o nome da definição do servidor atribuído ao servidor de banco de dados Informix na instrução CREATE SERVER.
- *vinnie* é o nome do proprietário ao qual a tabela, visualização ou sinônimo pertencem, a menos que o banco de dados seja ANSI-compliant. Em um banco de dados ANSI-compliant, é o nome do esquema.
- *japan* é o nome da tabela, visualização ou sinônimo remotos que você deseja acessar.

O servidor federado fecha os nomes dos esquemas e tabelas do Informix para maiúsculas, a menos que você coloque os nomes entre aspas.

Configurando acesso a origens de dados JDBC

Para configurar o servidor federado para acessar as origens de dados JDBC, é necessário fornecer ao servidor federado informações sobre as origens de dados e os objetos que deseja acessar.

Antes de Iniciar

As origens de dados acessadas por meio da API JDBC são referidas neste texto como origens de dados JDBC.

- O driver JDBC deve ser instalado e configurado no computador que age como o servidor federado.
- Verifique a configuração do servidor federado.
- Verifique o parâmetro federado para assegurar que a federação está ativada.
- Dependendo do seu cliente de origem de dados, você deve se certificar de que as seguintes variáveis sejam especificadas em ambas variáveis de ambiente e no arquivo `db2dj.ini`:
 - CLASSPATH system environment variable
 - Other system environment variables required by your data source client
 - A variável DB2LIBPATH do DB2 Profile Registry (db2set).

Para determinar quais variáveis são necessárias, consulte a documentação fornecida por seu cliente de origem de dados.

Por exemplo, se precisar especificar bibliotecas de tempo de execução, pode ser necessário especificar as seguintes variáveis de ambiente:

- LIBPATH em sistemas operacionais AIX.
- PATH em sistemas operacionais Windows.
- LD_LIBRARY_PATH em sistemas operacionais Linux e Solaris.

Nota: Se você configurar variáveis de ambiente do sistema ou modificar o arquivo `db2dj.ini`, você deve parar e reiniciar o InfoSphere Federation Server.

Sobre Esta Tarefa

Restrições

- O wrapper JDBC é suportado apenas no modo protegido.

- O wrapper JDBC não suporta as seguintes funções e instruções:
 - Instruções LOCK TABLE em apelidos
 - Isolamento de nível de instrução
- As origens de dados JDBC não suportam operações de atualização e exclusão posicionadas.
- O wrapper JDBC não suporta as instruções INSERT, UPDATE ou DELETE com relação às origens de dados que restringem o número de instruções de ação para cada conexão. Consulte a documentação da origem de dados para determinar se a origem de dados restringe o número de instruções de ação para cada conexão.
- O wrapper JDBC não suporta operações em tabelas que contêm colunas com tipos de dados que usam indicadores de tipo de dados SQL específicos ao driver. Os tipos de operações que não são suportadas incluem as instruções CREATE NICKNAME e SELECT em sessões de passagem. O wrapper JDBC suporta apenas os indicadores de tipo de dados de SQL definidos pela especificação do JDBC 3.0 e superior. Consulte a documentação do driver JDBC para obter as especificações do JDBC.
- O wrapper JDBC não suporta LOB em sessões de passagem.
- Restrições sobre o mapeamento de tipo e conversão do tipo de dados:
 - Tipos de dados não suportados: ARRAY, DATALINK, DISTINCT, JAVA_OBJECT, REF, STRUCT e OTHER
 - Tipos de dados com suporte limitado:
 - O suporte para o tipo de dado XML é limitado. O servidor federado processará um tipo de dado CLOB apenas se o tipo de dado JDBC relacionado for um tipo de dado CLOB ou SQLXML (JDBC 4.0). Caso contrário, não haverá suporte para o tipo de dado XML.
 - Os tipos de dados DBCS e UNICODE são armazenados pelo wrapper JDBC como UCS-2.
 - The DECFLOAT data type in DB2 databases and the NUMBER data type in Oracle databases have a larger scope; and their formats can be different to the corresponding JDBC wrapper data types. O mapeamento para um tipo de dado DECFLOAT ou NUMBER pode causar resultados imprecisos.

Procedimento

1. Prepare o servidor federado para acessar origens de dados através do JDBC.
2. Registre o wrapper JDBC.
3. Registre as definições de servidor para uma origem de dados JDBC.
4. Crie um mapeamento de usuário para uma origem de dados JDBC.
5. Teste a conexão com o servidor da origem de dados JDBC.
6. Registre apelidos para tabelas e visualizações da origem de dados JDBC.

Preparando o Servidor Federado para Acessar Origens de Dados através do JDBC

O servidor federado deve ser capaz de acessar as origens de dados JDBC. Para preparar o servidor federado, é necessário determinar se é necessário configurar a variável de ambiente CLASSPATH.

Antes de Iniciar

Se você usar um driver JDBC diferente do driver JDBC padrão do DB2 no arquivo db2jcc.jar, poderá precisar incluir as informações do driver JDBC na variável de ambiente CLASSPATH. Opcionalmente, você pode especificar os pacotes do driver

JDBC com o parâmetro DRIVER_PACKAGE da instrução CREATE SERVER, ao registrar as definições do servidor.

Procedimento

Registre o arquivo .jar do Java que contém o driver JDBC na variável de ambiente CLASSPATH.

Opção	Descrição
Para Linux e UNIX	<p>Execute o comando de exportação para registrar o driver JDBC. Por exemplo, se você especificar o driver JDBC do DB2, execute o seguinte comando:</p> <pre>export CLASSPATH=\$CLASSPATH:db2_instance_dir/sql1lib/java/db2jdcc.jar</pre> <p>em que <i>db2_instance_dir</i> é caminho de arquivo completo de onde a instância do sistema de banco de dados do DB2 está instalada.</p>
Para Windows	<p>Configure a variável de ambiente do sistema CLASSPATH como o driver JDBC:</p> <ol style="list-style-type: none"> 1. Efetue logon como administrador. 2. Abra o Painel de Controle e navegue para Sistema > Variáveis de Ambiente. 3. Inclua o nome do arquivo e o diretório do driver JDBC na variável do sistema CLASSPATH. Use um ponto e vírgula para separar a nova entrada de todas as entradas existentes. <p>Consulte a documentação do driver JDBC para obter informações específicas sobre como registrar as variáveis de ambiente do sistema.</p>

O que Fazer Depois

Após concluir essa tarefa, você deverá registrar o wrapper.

Registrando o Wrapper JDBC

É necessário registrar um wrapper para acessar origens de dados JDBC. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Procedimento

Utilize um dos seguintes métodos:

Método	Descrição
Run the CREATE WRAPPER statement from the command line and specify the default name for the JDBC wrapper.	<p>Por Exemplo:</p> <pre>CREATE WRAPPER JDBC;</pre> <p>Ao registrar um wrapper usando o nome padrão, JDBC, o servidor federado utiliza automaticamente a biblioteca do wrapper JDBC apropriada para o sistema operacional no qual o servidor federado está em execução.</p>

Método	Descrição
Run the CREATE WRAPPER statement from the command line and specify an alternative name for the JDBC wrapper.	<p>Se o nome padrão do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, você poderá substituir o nome padrão do wrapper por um de sua escolha. Se você não utilizar o nome padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.</p> <p>Por exemplo, para registrar um wrapper com o nome <i>jdbc_Wrapper</i> em um servidor federado que usa o AIX, execute a seguinte instrução:</p> <pre>CREATE WRAPPER jdbc_Wrapper LIBRARY 'libdb2rcjdbc.a';</pre> <p>O arquivo da biblioteca do wrapper que você especifica depende do sistema operacional do servidor federado.</p>

O que Fazer Depois

Após concluir essa tarefa, você deverá registrar as definições de servidor.

Arquivos da Biblioteca do Wrapper JDBC

Os arquivos da biblioteca do wrapper JDBC são incluídos no servidor federado ao instalar o wrapper.

Ao instalar o wrapper JDBC, os arquivos da biblioteca são incluídos no caminho do diretório padrão. Por exemplo, se o servidor federado for executado no AIX, os arquivos da biblioteca do wrapper incluídos no caminho do diretório serão *libdb2rcjdbc.a*, *libdb2rcjdbcF.a*, *libdb2rcjdbcU.a* e *db2qgjdbc.jar*. O arquivo da biblioteca do wrapper padrão é *libdb2rcjdbc.a*. Os outros arquivos da biblioteca do wrapper são usados internamente pelo wrapper JDBC.

Se você não usa o nome do wrapper padrão ao registrar um wrapper, é necessário especificar o parâmetro **LIBRARY** na instrução CREATE WRAPPER.

Use os seguintes caminhos do diretório padrão e os nomes do arquivo da biblioteca do wrapper para especificar o parâmetro **LIBRARY** na instrução CREATE WRAPPER:

Tabela 24. Caminhos do Diretório da Biblioteca do Cliente do JDBC e Nomes do Arquivo

Sistema Operacional	Caminho do Diretório	Arquivo da Biblioteca do Wrapper
AIX	<i>/usr/opt/install_path/lib32/</i> <i>/usr/opt/install_path/lib64/</i>	<i>libdb2rcjdbc.a</i>
Linux	<i>/opt/IBM/db2/install_path/lib32</i> <i>/opt/IBM/db2/install_path/lib64</i>	<i>libdb2rcjdbc.so</i>
Solaris	<i>/opt/IBM/db2/install_path/lib32</i> <i>/opt/IBM/db2/install_path/lib64</i>	<i>libdb2rcjdbc.so</i>
Windows	<i>%DB2PATH%\bin</i>	<i>db2rcjdbc.dll</i>

em que *install_path* é o caminho do diretório onde a federação está instalada no Linux ou UNIX.

Instrução CREATE WRAPPER - Exemplo para o Wrapper JDBC

Use a instrução CREATE WRAPPER para registrar o wrapper JDBC.

Para registrar um wrapper com o nome padrão em um servidor federado, execute a instrução CREATE WRAPPER com o nome do wrapper JDBC, por exemplo:

```
CREATE WRAPPER JDBC;
```

Nos exemplos a seguir, *jdbc_Wrapper* é o nome alternativo designado para o wrapper registrado no banco de dados federado.

Servidor Federado Linux e Solaris

O exemplo a seguir mostra como registrar um wrapper com um nome alternativo:

```
CREATE WRAPPER jdbc_Wrapper LIBRARY 'libdb2rcjdbc.so';
```

Servidor Federado AIX

O exemplo a seguir mostra como registrar um wrapper com um nome alternativo:

```
CREATE WRAPPER jdbc_Wrapper LIBRARY 'libdb2rcjdbc.a';
```

Servidor Federado Windows

O exemplo a seguir mostra como registrar um wrapper com um nome alternativo:

```
CREATE WRAPPER jdbc_Wrapper LIBRARY 'db2rcjdbc.dll';
```

Registrando Definições de Servidor para Origens de Dados JDBC

É necessário registrar cada servidor JDBC que deseja acessar no banco de dados federado.

Antes de Iniciar

Se você usar um driver JDBC diferente do driver JDBC padrão do servidor DB2, poderá precisar configurar a variável de ambiente CLASSPATH para especificar os arquivos do pacote do driver JDBC. Opcionalmente, você poderá especificar os pacotes do driver JDBC na instrução CREATE SERVER com a opção do servidor DRIVER_PACKAGE.

Sobre Esta Tarefa

Para registrar o servidor JDBC, é necessário especificar o nome do pacote do driver JDBC na biblioteca do driver JDBC e cadeia de conexão de JDBC do servidor remoto.

Procedimento

Run the CREATE SERVER statement from the command line for every JDBC server that you want to register,

Por Exemplo:

```
CREATE SERVER server_Definition_Name  
  TYPE jdbc_Datasource_Type  
  VERSION version_Number  
  WRAPPER jdbc_Wrapper_Name  
  OPTIONS (  
    DRIVER_CLASS 'jdbc_Driver_Class_Path',  
    URL 'jdbc_Url_Connect_String');
```

Importante: Ao executar a instrução `CREATE SERVER`, na verdade, ela não criará uma conexão com a origem de dados até que a instrução `CREATE NICKNAME` seja executada. Se você especificar as informações de conexão incorretas no parâmetro `OPTIONS`, você não será notificado do erro até que as instruções `CREATE NICKNAME` ou `pass-through` sejam executadas.

O que Fazer Depois

Após concluir essa tarefa, você deverá criar um mapeamento de usuário.

Instrução `CREATE SERVER` - Exemplos do Wrapper JDBC

Use a instrução `CREATE SERVER` para registrar definições de servidor para o wrapper JDBC. Este exemplo fornece os parâmetros necessários e um exemplo com parâmetros do servidor adicionais.

O exemplo a seguir mostra como registrar uma definição de servidor para uma origem de dados DB2 emitindo a instrução `CREATE SERVER`:

```
CREATE SERVER jdbc_server1
  TYPE JDBC
  VERSION 3.0
  WRAPPER jdbc_wrapper1
  OPTIONS (
    DRIVER_PACKAGE '/home/My_LIB/JDBC_driver/derbyclient.jar',
    DRIVER_CLASS 'com.ibm.db2.jcc.DB2Driver',
    URL 'jdbc:db2://server.example.com:50471/testdb');
```

Valores de parâmetros

jdbc_server1

Especifica um nome designado para o servidor de origem de dados JDBC. Nomes de definição de servidor duplicados não são permitidos.

TYPE JDBC

Especifica o tipo de servidor de origem de dados que você deseja acessar. Este parâmetro é opcional.

VERSION 3.0

Especifica a versão da origem de dados JDBC que você deseja acessar. Este parâmetro é opcional.

WRAPPER jdbc_wrapper1

Especifica o nome do wrapper especificado na instrução `CREATE WRAPPER`.

DRIVER_PACKAGE '/home/My_LIB/JDBC_driver/derbyclient.jar'

Especifica os pacotes do driver JDBC.

DRIVER_CLASS 'com.ibm.db2.jcc.DB2Driver'

Especifica a biblioteca do driver JDBC.

URL 'jdbc:db2://matthaus.cn.ibm.com:50471/testdb'

Especifica a cadeia de conexão do JDBC do servidor remoto.

Parâmetros do Servidor

Ao criar a definição do servidor, você pode especificar opções de servidor adicionais na instrução `CREATE SERVER`. As opções do servidor podem incluir as opções gerais do servidor e os parâmetros do servidor específicos ao JDBC.

Em geral, as configurações padrão para os parâmetros do servidor têm funcionalidade limitada. É possível usar os parâmetros do servidor para otimizar a configuração.

Para acessar as origens de dados JDBC, é necessário especificar os parâmetros do servidor DRIVER_CLASS e URL na instrução CREATE SERVER. Os parâmetro do servidor DRIVER_PACKAGE e JDBC_LOG são opcionais. A sintaxe do parâmetro OPTIONS a seguir especifica todos os parâmetros do servidor específicos ao JDBC:

```
OPTIONS (  
  DRIVER_PACKAGE '/path1/file1.jar: /path2/file2.jar',  
  DRIVER_CLASS 'com.ibm.db2.jcc.DB2Driver',  
  URL 'jdbc:db2://server.example.com:50471/testdb',  
  JDBC_LOG 'Y');
```

Parâmetros

DRIVER_PACKAGE '/path1/file1.jar: /path2/file2.jar'

Especifica os pacotes do driver JDBC e configura a variável de ambiente CLASSPATH.

DRIVER_CLASS 'com.ibm.db2.jcc.DB2Driver'

Especifica a biblioteca do driver JDBC do DB2.

URL 'jdbc:db2://server.example.com:50471/testdb'

Especifica a cadeia de conexão de JDBC que consiste em três partes que são todas separadas por dois pontos:

- O protocolo do banco de dados
- O nome do tipo de banco de dados ou o nome do driver de conectividade
- A identidade do banco de dados através de um alias ou subnome

Exemplos

Para bancos de dados DB2

```
jdbc:db2://server.example.com:50471/testdb
```

em que jdbc é o protocolo, db2 é o tipo de banco de dados e //server.example.com:50471/testdb é o alias do banco de dados que se refere a uma entrada do catálogo do banco de dados DB2 no cliente DB2.

Para Oracle

```
jdbc:oracle:thin:@//myhost:1521/orcl
```

em que jdbc é o protocolo, oracle é o tipo de banco de dados e thin:@//myhost:1521/orcl são as informações do cliente JDBC do Oracle para acessar o servidor Oracle.

JDBC_LOG 'Y'

Especifica como criar arquivos de log para rastreamento de erro. O valor padrão desta opção do servidor é N.

Exemplo

```
CREATE SERVER jdbc_server1  
  TYPE JDBC  
  VERSION 3.0  
  WRAPPER jdbc_wrapper1  
  OPTIONS (  
    DRIVER_PACKAGE '/home2/JDBC_driver/derbyclient.jar',  
    DRIVER_CLASS 'org.apache.derby.jdbc.ClientDriver',  
    URL 'jdbc:derby://9.181.139.129:1527/testdb9;create=true;',  
    JDBC_LOG 'Y');
```

Criando Mapeamentos de Usuário para Origens de Dados JDBC

É necessário definir uma associação (um mapeamento de usuário) entre cada ID do usuário do servidor federado e o ID do usuário da origem de dados correspondente.

Sobre Esta Tarefa

Ao tentar acessar um servidor JDBC, o servidor federado estabelece uma conexão com o servidor JDBC usando um ID do usuário e senha para essa origem de dados.

Crie um mapeamento de usuário para cada ID do usuário que acesse o sistema federado para enviar pedidos distribuídos à origem de dados JDBC.

Procedimento

Execute a instrução CREATE USER MAPPING para mapear um ID do usuário local para o ID do usuário e senha da origem de dados JDBC:

```
CREATE USER MAPPING FOR local_userID
  SERVER server_definition_name
  OPTIONS (
    REMOTE_AUTHID 'remote_userID',
    REMOTE_PASSWORD 'remote_password');
```

Os parâmetros de mapeamento de usuário **REMOTE_AUTHID** e **REMOTE_PASSWORD** são necessários.

O que Fazer Depois

Após concluir essa tarefa, você poderá testar a conexão com a origem de dados JDBC.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper JDBC

Este exemplo mostra como usar o USER de registro especial do DB2 com a instrução CREATE USER MAPPING.

O exemplo a seguir mostra como mapear um ID de autorização federado para um ID do usuário e senha da origem de dados JDBC:

```
CREATE USER MAPPING FOR arturo
  SERVER jdbc_server1
  OPTIONS (
    REMOTE_AUTHID 'art',
    REMOTE_PASSWORD 'red4blue');
```

Parâmetros

SERVER arturo

Especifica o ID de autorização local mapeado para o ID do usuário e senha remotos, que estão definidos na origem de dados JDBC.

OPTIONS jdbc_server1

Especifica o nome de definição do servidor definido na instrução CREATE SERVER para a origem de dados JDBC.

REMOTE_AUTHID 'art'

Especifica o ID do usuário remoto para o qual você mapeia arturo. O

valor faz distinção entre maiúsculas e minúsculas, a não ser que você configure o parâmetro do servidor FOLD_ID como U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD 'red4blue'

Especifica a senha remota associada a 'art'. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

USER de Registro Especial do DB2

É possível usar o USER de registro especial do DB2 para mapear o ID de autorização da pessoa que emite a instrução CREATE USER MAPPING para o ID de autorização da origem de dados que está especificado na opção de usuário REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER
  SERVER jdbc_server1
  OPTIONS (
    REMOTE_AUTHID 'art',
    REMOTE_PASSWORD 'red4blue');
```

Testando Conexões com Servidores de Origem de Dados JDBC

É possível testar a conexão com o servidor de origem de dados JDBC usando a definição de servidor e os mapeamentos de usuário definidos.

Procedimento

Abra uma sessão de passagem e emita uma instrução SELECT nas tabelas do sistema da origem de dados JDBC. Se a instrução SELECT retornar uma contagem, a definição do servidor e o mapeamento de usuário estão configurados apropriadamente.

```
SET PASSTHRU server_definition_name
SELECT count(*) FROM schema_name.table_name
SET PASSTHRU RESET
```

Se a instrução SELECT retornar um erro, resolva os problemas de erros de conexão.

O que Fazer Depois

Após concluir essa tarefa, você deverá registrar os apelidos para as tabelas e visualizações da origem de dados JDBC.

Resolução de Problemas de Erros de Conexão da Origem de Dados

Uma conexão de teste com o servidor de origem de dados pode retornar um erro por diversas razões. Há ações que você pode executar para determinar a causa de um erro.

Sintoma

É retornado um erro quando você tentar se conectar à origem de dados.

Causa

Há diversas causas possíveis para um problema de conexão.

Solução do Problema

Para resolver problemas de erros de conexão com a origem de dados, verifique os seguintes itens:

- Verifique se a origem de dados está disponível.
- Se aplicável, certifique-se de que o servidor de origem de dados esteja configurado para conexões de chegada.
- Certifique-se de suas configurações de mapeamento de usuário para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com a origem de dados. Altere o mapeamento de usuário ou crie outro, conforme necessário.
- Se aplicável, certifique-se de que o software do cliente da origem de dados no servidor federado esteja instalado e configurado corretamente para conexão com a origem de dados.
- Para origens de dados ODBC, certifique-se de que o driver ODBC no servidor federado esteja instalado e configurado corretamente para conexão com o servidor de origem de dados ODBC. Em servidores federados que executam Windows, utilize a ferramenta Administrador de Origem de Dados ODBC para verificar o driver. Em servidores de dados que executam UNIX, consulte a documentação do fornecedor do cliente ODBC.
- Verifique se as configurações para as variáveis configuradas no servidor federado estão corretas para a origem de dados. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis no arquivo db2dj.ini e as variáveis do DB2 Profile Registry (db2set).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e a crie novamente.

Registrando Apelidos para Tabelas e Visualizações da Origem de Dados JDBC

Para cada definição do servidor JDBC registrada, é necessário registrar um apelido para cada tabela ou visualização a ser acessada. Use esses apelidos, em vez dos nomes dos objetos de origem de dados ao consultar as origens de dados JDBC.

Antes de Iniciar

Atualize as estatísticas na origem de dados JDBC antes de registrar um apelido. O banco de dados federado confia nas estatísticas do catálogo da origem de dados para otimizar o processamento da consulta. Utilize o comando da origem de dados equivalente ao comando do DB2 RUNSTATS para atualizar as estatísticas da origem de dados.

Procedimento

Emita a instrução CRIAR APELIDO a partir da linha de comandos.

Por Exemplo:

```
CREATE NICKNAME nickname  
FOR server_definition_name."remote_schema"."remote.table" ;
```

Os apelidos podem ter até 128 caracteres de comprimento. Ao criar o apelido, o servidor federado consulta o catálogo de origem de dados. Esta consulta testa a conexão com a tabela ou visualização da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

O que Fazer Depois

Repita essa etapa para cada tabela ou visualização do JDBC para a qual deseja criar um apelido.

Instrução CREATE NICKNAME - Exemplos para o Wrapper JDBC

O exemplo mostra como registrar um apelido para a tabela ou visualização JDBC usando a instrução CREATE NICKNAME.

Essa instrução especifica a definição de servidor e o esquema e a tabela remotos:

```
CREATE NICKNAME cust_europe FOR jdbc_server."vinnie"."italy"
```

cust_europe

Um apelido exclusivo usado para identificar a tabela ou a visualização do JDBC. O apelido deve ser exclusivo dentro do esquema.

Importante: O apelido é um nome de duas partes; o esquema e o apelido. Se você omitir o esquema ao registrar o apelido, o esquema do apelido será configurado como o ID de autorização do usuário que registra o apelido.

Se a origem de dados JDBC não suportar esquemas, omita o esquema a partir da instrução CREATE NICKNAME, por exemplo:

```
CREATE NICKNAME cust_europe FOR jdbc_server."italy"
```

jdbc_server."vinnie"."italy"

Um identificador de três partes para o objeto remoto:

jdbc_server

O nome da definição do servidor que você designou para o servidor de origem de dados do JDBC na instrução CREATE SERVER.

vinnie O ID do usuário do proprietário ao qual a tabela ou a visualização pertence.

italy O nome da tabela ou visualização remota que você deseja acessar.

O servidor federado desdobra os nomes dos esquemas e tabelas do JDBC para maiúsculas, a não ser que os nomes sejam colocados entre aspas duplas.

Configurando Acesso às Origens de Dados do Microsoft SQL Server

Para configurar o servidor federado para acessar origens de dados do Microsoft SQL Server, você deve fornecer ao servidor federado informações sobre as origens de dados e objetos que deseja acessar.

Antes de Iniciar

- O driver ODBC deve estar instalado e configurado no servidor federado.
- A federação deve ser instalada em um servidor que atue como o servidor federado.
- Verifique a configuração do servidor federado.

- Verifique o parâmetro federado para assegurar que a federação está ativada.

Procedimento

1. Utilize um dos seguintes métodos para preparar o servidor federado e o banco de dados federado, dependendo do seu sistema operacional.
 - Prepare o servidor federado e o banco de dados federado (Windows).
 - Prepare o servidor federado e o banco de dados federado (UNIX)
2. Configure as variáveis de ambiente para o wrapper do Microsoft SQL Server.
3. Registre o wrapper.
4. Registre a definição do servidor.
5. Crie os mapeamentos de usuário.
6. Teste a conexão com o servidor remoto do Microsoft SQL Server
7. Registre apelidos para as tabelas e visualizações do Microsoft SQL Server.

Preparando o Servidor Federado para Acessar as Origens de Dados Microsoft SQL Server (Windows)

Em servidores federados que executam Windows, o servidor federado deve ser capaz de acessar origens de dados do Microsoft SQL Server. Para preparar o servidor federado, você deve verificar as configurações no DSN do Sistema ODBC e testar a conexão com as origens de dados do do Microsoft SQL Server.

Procedimento

1. Verifique se o DSN do Sistema ODBC está configurado para conectar-se à origem de dados do Microsoft SQL Server. No Painel de Controle, localize a entrada DSN existente para o servidor remoto do Microsoft SQL Server ou crie uma entrada DSN.

A entrada DSN para o servidor remoto do Microsoft SQL Server é o valor que você utilizará para a opção de servidor NODE quando registrar a definição de servidor no banco de dados federado.
2. Utilize um dos seguintes métodos para testar a conexão com a origem de dados do Microsoft SQL Server:
 - Selecione **Configurar** na janela Administrador da Origem de Dados ODBC.
 - Utilize a ferramenta de consulta do Microsoft SQL Server.

O que Fazer Depois

Depois de concluir essa tarefa, é possível configurar as variáveis de ambiente.

Preparando o Servidor Federado para Acessar Origens de Dados do Microsoft SQL Server (Linux, UNIX)

Em servidores federados que executam Linux ou UNIX, o servidor federado deve ser capaz de acessar as origens de dados Microsoft SQL Server. Para preparar o servidor federado, você deverá verificar as configurações no arquivo `odbc.ini`, criar links simbólicos e testar a conexão com as origens de dados Microsoft SQL Server.

Procedimento

1. Verifique se o arquivo `odbc.ini` está atualizado no servidor federado. Se o arquivo `odbc.ini` não existir no servidor federado, será possível criá-lo em um editor de texto. Consulte a documentação do fornecedor do cliente ODBC para obter informações sobre o arquivo `odbc.ini`.

Lembre-se: Coloque o arquivo `odbc.ini` ou uma cópia deste arquivo no diretório inicial do proprietário da instância do DB2 para garantir que ele possa ser acessado se o proprietário da instância não for o usuário `root`.

- Verifique se o caminho para o `odbc.ini` está na variável de ambiente `ODBCINI`. A partir de um prompt de comandos do sistema operacional, emita o seguinte comando:

```
export ODBCINI=$HOME/.odbc.ini
```

- Crie os links simbólicos apropriados:

Sistema operacional do servidor federado	Etapa
Linux	<p>Crie os seguintes links simbólicos dependendo do driver ODBC que for instalado, por exemplo:</p> <pre>ln -s \$DJX_ODBC_LIBRARY_PATH/.. /locale/usr/local/locale</pre> <pre>ln -s \$DJX_ODBC_LIBRARY_PATH/libodbcinst.so/ usr/lib/libodbcinst.so</pre> <p>Exemplos</p> <ul style="list-style-type: none"> Para o driver ODBC do IBM Branded DataDirect: <ul style="list-style-type: none"> Crie links simbólicos para a versão 5.2, da seguinte forma: <pre>ln -s \$DJX_ODBC_LIBRARY_PATH/libVMicu22.so /usr/lib/libVMicu22.so</pre> Crie o link simbólico para a versão 5.3, da seguinte forma: <pre>ln -s \$DJX_ODBC_LIBRARY_PATH/libVMicu23.so /usr/lib/libVMicu23</pre> Para o DataDirect Technologies Connect para driver ODBC: <ul style="list-style-type: none"> É possível criar o link simbólico para a versão 4.2, da seguinte forma: <pre>ln -s \$DJX_ODBC_LIBRARY_PATH/libivicu19.so/ usr/lib/libivicu19.so</pre> Crie o link simbólico para a versão 5.0, da seguinte forma: <pre>ln -s \$DJX_ODBC_LIBRARY_PATH/libivicu20.so/ usr/lib/libivicu20.so</pre> <p>Se você não criar o link simbólico, a instrução <code>CREATE WRAPPER MSSQLODBC3</code> poderá falhar com a seguinte mensagem de erro:</p> <pre>SQL10013N O nome da biblioteca especificada não pôde ser carregado.</pre>
Solaris	<p>Crie o seguinte link simbólico:</p> <pre>ln -s \$DJX_ODBC_LIBRARY_PATH/.. locale \$HOME/sqllib/locale</pre> <p>em que <code>\$HOME</code> é o diretório inicial do proprietário da instância do DB2.</p>

A tabela a seguir lista o nome da biblioteca para cada versão do driver ODBC.

Tabela 25. ODBC driver versions

Versão do driver ODBC	Nome da biblioteca
4.2	libodbcinst.so libVMicu19.so
5.0	libodbcinst.so libVMicu20.so
5.1	libodbcinst.so libVMicu21.so
5.3	libodbcinst.so libVMicu23.so
6.0	libodbcinst.so libVMicu24.so

4. Instale os drivers ODBC. Para obter informações sobre como instalar os drivers ODBC, consulte *Instalando drivers ODBC*.

O que Fazer Depois

Depois de concluir essa tarefa, é possível configurar as variáveis de ambiente.

Configurando as Variáveis de Ambiente do Microsoft SQL Server

As variáveis de ambiente do Microsoft SQL Server devem ser configuradas no arquivo `db2dj.ini` no servidor federado.

Sobre Esta Tarefa

Restrições

Revise as restrições para o arquivo `db2dj.ini` antes de iniciar esta tarefa.

O arquivo `db2dj.ini` contém informações de configuração sobre o driver ODBC do Microsoft SQL Server que é instalado no seu servidor federado.

Há variáveis de ambiente requeridas e opcionais para as origens de dados do Microsoft SQL Server.

Se você instalou o software do cliente Microsoft SQL Server antes de ter instalado o wrapper do Microsoft SQL Server, as variáveis de ambiente do Microsoft SQL Server requeridas estão configuradas no arquivo `db2dj.ini`.

Você deve configurar as variáveis de ambiente utilizando as etapas nesta tarefa se não instalou o software do cliente do Microsoft SQL Server antes de ter instalado o wrapper do Microsoft SQL Server ou se desejar configura qualquer uma das variáveis de ambiente opcionais.

Procedimento

1. Utilize um dos seguintes métodos:

Método	Etapa
Configure as variáveis de ambiente com a ferramenta de configuração do wrapper	Execute a ferramenta de configuração do wrapper para configurar as variáveis de ambiente necessárias e as opcionais. Siga as instruções em Executando a ferramenta de configuração do wrapper.
Configurar manualmente as variáveis de ambiente.	<p>Edite o arquivo <code>db2dj.ini</code>.</p> <ul style="list-style-type: none">• Em servidores federados que executam UNIX, este arquivo está localizado no diretório <code>sql1lib/cfg</code>.• Em servidores federados que executam Windows, este arquivo está localizado no diretório <code>%DB2PATH%\cfg</code>. <p>Se o arquivo não existir, você poderá criar um arquivo com o nome <code>db2dj.ini</code> utilizando qualquer editor de texto. No arquivo <code>db2dj.ini</code>, você deve especificar o caminho completo no valor das variáveis de ambiente; caso contrário, encontrará erros.</p> <p>Por exemplo:</p> <pre>DJX_ODBC_LIBRARY_PATH=/opt/odbc/lib ODBCINI=/opt/odbc/.odbc.ini</pre>

2. Para garantir que as variáveis de ambiente estejam configuradas no servidor federado, recicle a instância do DB2 com os seguintes comandos:

```
db2stop
db2start
```

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar o wrapper.

Variáveis de Ambiente do Microsoft SQL Server

Há variáveis de ambiente requeridas e opcionais para as origens de dados do Microsoft SQL Server. Essas variáveis são definidas no arquivo `db2dj.ini`.

As seguintes variáveis de ambiente são válidas para o Microsoft SQL Server:

- `DJX_ODBC_LIBRARY_PATH`
- `ODBCINI`
- `LD_LIBRARY_PATH` (apenas Solaris)

Descrições da Variável

`DJX_ODBC_LIBRARY_PATH`

Especifica o caminho do diretório para os arquivos de biblioteca do ODBC. Esta variável também deve ser especificada nos servidores federados que executam o Solaris.

Por exemplo:

```
DJX_ODBC_LIBRARY_PATH=ODBC_driver_directory/lib
```

`ODBC_driver_directory` é o caminho do diretório no qual o driver ODBC está instalado.

ODBCINI

Especifica o caminho do diretório no qual seu arquivo de configuração do ODBC (`odbc.ini`) está localizado.

Por exemplo:

```
ODBCINI=/home/db2inst1/.odbc.ini
```

Não configure a variável de ambiente ODBCINI como uma variável do sistema.

LD_LIBRARY_PATH (apenas Solaris)

Em servidores federados que executam Solaris, especifica o caminho do diretório para os arquivos da biblioteca do ODBC.

Por exemplo:

```
LD_LIBRARY_PATH=ODBC_driver_directory/lib
```

Registrando o Wrapper do Microsoft SQL Server

Você deve registrar um wrapper para acessar as origens de dados do Microsoft SQL Server. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Os wrappers são implementados como um conjunto de arquivos de biblioteca

Procedimento

Emita a instrução `CREATE WRAPPER` a partir da linha de comandos e especifique o nome padrão para o wrapper do Microsoft SQL Server.

Por exemplo:

```
CREATE WRAPPER MSSQLODBC3;
```

Lembre-se: Quando você registrar o wrapper utilizando o nome padrão, `MSSQLODBC3`, o servidor federado utiliza automaticamente a biblioteca do wrapper do Microsoft SQL Server apropriada para o sistema operacional no qual seu servidor federado está sendo executado.

Se o nome padrão do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, você poderá substituir o nome padrão do wrapper por um de sua escolha. Se você não utilizar o nome padrão, deverá incluir o parâmetro `LIBRARY` na instrução `CREATE WRAPPER`.

Por exemplo, para registrar um wrapper com o nome `sqlserver_wrapper` em um servidor federado que utilize AIX, emita a seguinte instrução:

```
CREATE WRAPPER sqlserver_wrapper  
LIBRARY 'libdb2mssql3.a';
```

O arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado.

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar a definição do servidor.

Arquivos da Biblioteca do Wrapper do Microsoft SQL Server

Os arquivos da biblioteca do wrapper do Microsoft SQL Server são incluídos no servidor federado quando você instalar o wrapper.

Ao instalar o wrapper do Microsoft SQL Server, três arquivos de biblioteca são incluídos no caminho de diretório padrão. Por exemplo, se o servidor federado está em execução no AIX, os arquivos da biblioteca do wrapper que estão incluídos

no caminho do diretório são libdb2mssql3.a, libdb2mssql3F.a e libdb2mssql3U.a. O arquivo da biblioteca do wrapper padrão é libdb2mssql3.a. Os outros arquivos da biblioteca do wrapper são utilizados internamente pelo wrapper do Microsoft SQL Server.

Se não utilizar o nome padrão do wrapper ao registrá-lo, você deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER e especificar o nome do arquivo da biblioteca padrão do wrapper.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 26. Locais de Biblioteca e Nomes de Arquivos do Cliente Microsoft SQL Server

Sistema Operacional	Caminho do Diretório	Nome do Arquivo de Biblioteca
AIX	/usr/opt/install_path/lib32/ /usr/opt/install_path/lib64/	libdb2mssql3.a
Linux	/opt/IBM/db2/install_path/lib32 /opt/IBM/db2/install_path/lib64	libdb2mssql3.so
Solaris	/opt/IBM/db2/install_path/lib32 /opt/IBM/db2/install_path/lib64	libdb2mssql3.so
Windows	%DB2PATH%\bin	db2mssql3.dll

install_path é o caminho do diretório em que servidor federado está instalado no UNIX ou Linux.

Registrando as Definições do Servidor para uma Origem de Dados do Microsoft SQL Server

Você deve registrar cada servidor remoto do Microsoft SQL Server que deseja acessar no banco de dados federado.

Procedimento

1. Localize o nome do nó para o Microsoft SQL Server.
 - Em servidores federados que executam o Windows, o nome do nó é o nome do DSN do Sistema que você especificou para o servidor remoto do Microsoft SQL Server que está acessando.
 - Em servidores federados que executam o UNIX, o nome do nó é definido no arquivo `.odbc.ini`.

No início do arquivo `.odbc.ini`, há seção denominada ODBC Data Sources, que lista os nós. Cada um dos nós possui uma seção no arquivo `.odbc.ini` que descreve o nó.

O exemplo a seguir é um arquivo `.odbc.ini` no AIX. Os nomes dos nós são `[rawilson]` e `[medusa]`.

```
[ODBC Data Sources]
rawilson=MS SQL Server 2000
medusa=MS SQL Server 2000
[rawilson]
Driver=/opt/odbc/lib/ddmsss20.so
Description=MS SQL Server Driver for AIX
Address=9.112.30.39,1433
[medusa]
Driver=/opt/odbc/lib/ddmsss20.so
```



```
Description=MS SQL Server Driver for AIX
Address=9.112.98.123,1433
[ODBC]
InstallDir=/opt/odbc
```

2. Emita a instrução CREATE SERVER a partir da linha de comandos para criar a definição de servidor.

Por exemplo:

```
CREATE SERVER server_definition_name
TYPE MSSQLSERVER
VERSION version_number
WRAPPER wrapper_name
OPTIONS (NODE 'node_name',
         DBNAME 'database_name');
```

Embora as variáveis '*node_name*' e '*db_name*' sejam especificadas como opções na instrução CREATE SERVER, estas opções são requeridas para as origens de dados do Microsoft SQL Server.

Após o registro da definição do servidor, utilize a instrução ALTER SERVER para incluir ou descartar opções do servidor.

O que Fazer Depois

Após concluir esta tarefa, você pode criar mapeamentos de usuário.

Instrução CREATE SERVER - Exemplos para o Wrapper do Microsoft SQL Server

Utilize a instrução CREATE SERVER para registrar definições de servidor para o wrapper do Microsoft SQL Server. Este tópico fornece um exemplo completo com os parâmetros requeridos e um exemplo com opções de servidor adicionais.

O exemplo a seguir mostra como registrar uma definição de servidor para um wrapper do Microsoft SQL Server emitindo a instrução CREATE SERVER:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER VERSION 2000 WRAPPER wrapper_name
         OPTIONS (NODE 'sqlnode', DBNAME 'africa');
```

sqlserver

Um nome que você atribui ao servidor remoto do Microsoft SQL Server. Nomes de definição de servidor duplicados não são permitidos.

TYPE MSSQLSERVER

Especifica o tipo de origem de dados para o qual você está configurando acesso. Para o wrapper do Microsoft SQL Server, o tipo de servidor deve ser MSSQLSERVER.

VERSION 2000

A versão do servidor de banco de dados do Microsoft SQL Server que você deseja acessar.

WRAPPER *wrapper_name*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

NODE '*sqlnode*'

O nome do nó no qual reside o servidor de remoto do Microsoft SQL Server. Em servidores federados que executam Windows, o nome do DSN do Sistema para o servidor remoto do Microsoft SQL Server que você está acessando. Em servidores federados que executam o UNIX, o nó que é definido no arquivo .odbc.ini.

Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para as origens de dados do Microsoft SQL Server.

DBNAME *'africa'*

O nome do banco de dados do Microsoft SQL Server que você deseja acessar. Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do banco de dados seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para as origens de dados do Microsoft SQL Server.

Opções do Servidor

Quando você cria uma definição do servidor, é possível especificar opções adicionais do servidor na instrução CREATE SERVER. As opções do servidor podem ser opções do servidor gerais e opções do servidor específicas do Microsoft SQL Server.

A opção do servidor COLLATING_SEQUENCE especifica se a origem de dados utiliza a mesma seqüência de intercalação que o servidor federado ou uma seqüência de intercalação diferente. Em um servidor de banco de dados do Microsoft SQL Server que esteja executando o Windows NT ou Windows 2000, a seqüência de intercalação padrão não faz distinção entre letras maiúsculas e minúsculas (por exemplo, 'STEWART' e 'StewART' são considerados iguais). Para garantir os resultados corretos no servidor federado, configure a opção de servidor COLLATING_SEQUENCE como 'I'. Esta configuração indica que a origem de dados do Microsoft SQL Server não faz distinção entre letras maiúsculas e minúsculas.

O servidor federado não push down consultas se os resultados que são retornados das origens de dados forem diferentes dos resultados que são retornados ao processar a consulta no servidor federado. Ao configurar a opção do servidor COLLATING_SEQUENCE como 'I', o servidor federado não push down consultas que contêm dados ou expressões de cadeia e que também contêm algumas das seguintes cláusulas, predicados ou funções:

- cláusulas GROUP BY
- cláusulas DISTINCT
- Predicados básicos, tais como igual a (=)
- Funções agregadas, tais como MIN ou MAX

O exemplo a seguir mostra como especificar a opção do servidor COLLATING_SEQUENCE:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER VERSION 2000 WRAPPER mssqlodbc3
OPTIONS (NODE 'sqlnode', DBNAME 'africa', COLLATING_SEQUENCE 'I');
```

Criando os Mapeamentos de Usuário para uma Origem de Dados do Microsoft SQL Server

Ao tentar acessar um servidor remoto do Microsoft SQL Server, o servidor federado estabelece uma conexão com o servidor remoto do Microsoft SQL Server utilizando um ID de usuário e uma senha que são válidos para a origem de dados.

Procedimento

Emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR local_userID SERVER server_definition_name  
    OPTIONS (REMOTE_AUTHID 'remote_userID', REMOTE_PASSWORD 'remote_password');
```

O que Fazer Depois

Após concluir esta tarefa, você pode testar a conexão com as tabelas e visualizações do Microsoft SQL Server.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper do Microsoft SQL Server

Utilize a instrução CREATE USER MAPPING para mapear um ID de autorização federado para um ID de usuário e senha do Microsoft SQL Server remoto. Este tópico fornece um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o USER de registro especial do DB2 com a instrução CREATE USER MAPPING.

O exemplo a seguir mostra como mapear um ID de autorização federado para um ID do usuário e senha do Microsoft SQL Server remoto:

```
CREATE USER MAPPING FOR elizabeth SERVER sqlserver  
    OPTIONS (REMOTE_AUTHID 'liz', REMOTE_PASSWORD 'abc123');
```

elizabeth

Especifica o ID de autorização que você está mapeando para um ID de usuário e senha remotos, que estão definidos no servidor remoto Microsoft SQL Server.

SERVER *sqlserver*

Especifique o nome da definição do servidor registrado na instrução CREATE SERVER para o servidor remoto Microsoft SQL Server.

REMOTE_AUTHID '*liz*'

Especifica o ID do usuário para o qual você está mapeando *elizabeth*. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_ID para 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD '*abc123*'

Especifica a senha remota que está associada com '*liz*'. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

USER de Registro Especial do DB2

Você pode utilizar o USER de registro especial do DB2 para mapear o ID de autorização da pessoa que está emitindo a instrução CREATE USER MAPPING para o ID do usuário remoto que está especificado na opção REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER SERVER sqlserver  
    OPTIONS (REMOTE_AUTHID 'liz', REMOTE_PASSWORD 'abc123');
```

Testando a Conexão com o Servidor Remoto Microsoft SQL Server

Teste a conexão com o servidor remoto Microsoft SQL Server para determinar se o servidor federado está configurado corretamente para acessar as origens de dados do Microsoft SQL Server.

Sobre Esta Tarefa

Você pode testar a conexão com o servidor remoto Microsoft SQL Server utilizando a definição do servidor e os mapeamentos de usuário definidos.

Procedimento

Abra uma sessão de passagem e emita uma instrução SELECT nas tabelas de sistema do Microsoft SQL Server. Se a instrução SELECT retornar uma contagem, a definição do servidor e o mapeamento de usuário estão configurados apropriadamente.

Por exemplo:

```
SET PASSTHRU server_definition_name
SELECT count(*) FROM dbo.sysobjects
SET PASSTHRU RESET
```

O que Fazer Depois

Após concluir esta tarefa, você pode registrar apelidos.

Resolução de Problemas de Erros de Conexão da Origem de Dados

Uma conexão de teste com o servidor de origem de dados pode retornar um erro por diversas razões. Há ações que você pode executar para determinar a causa de um erro.

Sintoma

É retornado um erro quando você tentar se conectar à origem de dados.

Causa

Há diversas causas possíveis para um problema de conexão.

Solução do Problema

Para resolver problemas de erros de conexão com a origem de dados, verifique os seguintes itens:

- Verifique se a origem de dados está disponível.
- Se aplicável, certifique-se de que o servidor de origem de dados esteja configurado para conexões de chegada.
- Certifique-se de suas configurações de mapeamento de usuário para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com a origem de dados. Altere o mapeamento de usuário ou crie outro, conforme necessário.
- Se aplicável, certifique-se de que o software do cliente da origem de dados no servidor federado esteja instalado e configurado corretamente para conexão com a origem de dados.

- Para origens de dados ODBC, certifique-se de que o driver ODBC no servidor federado esteja instalado e configurado corretamente para conexão com o servidor de origem de dados ODBC. Em servidores federados que executam Windows, utilize a ferramenta Administrador de Origem de Dados ODBC para verificar o driver. Em servidores de dados que executam UNIX, consulte a documentação do fornecedor do cliente ODBC.
- Verifique se as configurações para as variáveis configuradas no servidor federado estão corretas para a origem de dados. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis no arquivo `db2dj.ini` e as variáveis do DB2 Profile Registry (`db2set`).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e a crie novamente.

Registrando Apelidos para as Tabelas e Visualizações do Microsoft SQL Server

Para cada definição de servidor remoto do Microsoft SQL Server que você registrar, deverá registrar um apelido para cada tabela ou visualização que deseja acessar. Utilize este apelidos, ao invés dos nomes dos objetos de origem de dados, ao consultar os servidores remotos do Microsoft SQL Server.

Antes de Iniciar

Para garantir que o banco de dados federado possua estatísticas atuais e completas, execute o procedimento armazenado `sp_createstats` do Microsoft SQL Server e o comando **CREATE STATISTICS** do Microsoft SQL Server a partir do banco de dados Microsoft SQL Server antes de criar o apelido.

O procedimento armazenado `sp_createstats` reúne estatísticas sobre todas as colunas padrão em uma tabela em uma origem de dados do Microsoft SQL Server, mas não reúne estatísticas para colunas que aparecem primeiro em um índice. Para garantir que o banco de dados federado possua estatísticas completas sobre a tabela do Microsoft SQL Server, você também deve utilizar o comando **CREATE STATISTICS** do Microsoft SQL Server para reunir estatísticas para cada coluna que aparece primeiro em um índice.

Quando você utilizar o comando **CREATE STATISTICS** a partir do banco de dados do Microsoft SQL Server, você deve fornecer à estatística o mesmo nome que o da coluna para as quais as estatísticas estão sendo coletadas. Ao fornecer à estatística o mesmo nome que o da coluna, você garante que quando registrar o apelido com a instrução **CREATE NICKNAME**, o banco de dados federado lê as estatísticas coletadas pelo comando **CREATE STATISTICS** do Microsoft SQL Server.

Procedimento

Emita a instrução **CREATE NICKNAME** a partir da linha de comandos.

Por exemplo:

```
CREATE NICKNAME nickname
```

```
FOR server_definition_name."remote_schema"."remote.table" ;
```

Ao criar um apelido, o servidor federado consulta o catálogo de origem de dados utilizando o apelido. Esta consulta testa a conexão com a tabela, visualização ou sinônimo da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

O que Fazer Depois

Repita esta etapa para cada tabela ou visualização do Microsoft SQL Server para a qual você deseja criar um apelido.

Instrução CREATE NICKNAME - Exemplos para o Wrapper do Microsoft SQL Server

Utilize a instrução CREATE NICKNAME para registrar um apelido para uma tabela ou visualização do Microsoft SQL Server que você deseja acessar. Este tópico fornece um exemplo completo com os parâmetros necessários.

O exemplo a seguir mostra como registrar um apelido para uma tabela ou visualização do Microsoft SQL Server utilizando a instrução CREATE NICKNAME.

```
CREATE NICKNAME cust_africa FOR sqlserver."vinnie"."egypt"
```

cust_africa

Um apelido exclusivo que é utilizado para identificar a tabela ou visualização do Microsoft SQL Server.

Importante: O apelido é um nome de duas partes -o esquema e o apelido. Se você omitir o esquema ao registrar o apelido, o esquema do apelido será o ID de autorização do usuário que registrar o apelido.

sqlserver."vinnie"."egypt"

Um identificador de três partes para o objeto remoto:

- *sqlserver* é o nome da definição de servidor que você designou para o servidor remoto Microsoft SQL Server na instrução CREATE SERVER.
- *vinnie* é o ID do usuário do proprietário ao qual pertence a tabela ou visualização.
- *egypt* é o nome da tabela ou visualização remota que você deseja acessar.

O servidor federado converte os nomes dos esquemas e tabelas do Microsoft SQL Server em letras maiúsculas, a menos que você coloque os nomes entre aspas.

Utilizando Informações de Rastreamento do ODBC para Resolver Problemas de Conexões com as Origens de Dados do Microsoft SQL Server

Se você tiver problemas ao conectar-se à origem de dados, é possível obter informações de rastreamento do ODBC para analisar e resolver os problemas.

Sintoma

Entretanto, a ativação de um rastreamento impacta o desempenho do sistema. Você deve desativar o rastreamento após ter resolvido os problemas de conectividade.

Se você não conseguir se conectar à origem de dados com o wrapper do Microsoft SQL Server, a execução de um rastreamento pode ajudá-lo a diagnosticar o problema.

Causa

A causa do problema pode ser um erro na configuração do wrapper.

Diagnosticando o Problema

Para diagnosticar o problema em um servidor federado executando o Windows:

1. No Painel de Controle, abra a pasta **Ferramentas Administrativas**.
2. Clique em **Origens de Dados (ODBC)** para abrir a janela Administrador da Origem de Dados ODBC.
3. Clique na guia Rastreio.
4. Clique em **Iniciar Rastreio Agora** para iniciar o utilitário de rastreio.

Em um servidor federado executando UNIX:

1. Altere o arquivo `odbc.ini`.

Por exemplo, se você utilizar o driver DataDirect ODBC 3.x, localize o exemplo do arquivo `odbc.ini` no diretório do cliente. O arquivo `odbc.ini` contém uma amostra das configurações que são necessárias para ativar os arquivos de rastreio:

```
[ODBC]
Trace=1
TraceFile=/home/user1/trace_dir/filename.xxx
TraceDll==ODBC_driver_directory/odbc32.dll
InstallDir=/opt/odbc
```

Para ativar o rastreio, configure a primeira linha como `Trace=1`. Para desativar o rastreio, configure a primeira linha como `Trace=0`. O valor da configuração `TraceFile` é o caminho e o nome do arquivo para os quais a instância do banco de dados federado possui acesso de gravação.

Solução do Problema

Verifique o arquivo de log de rastreio para saber quais são os problemas.

No Windows, abra o Administrador da Origem de Dados ODBC e clique na guia Rastreio. O caminho para o arquivo de log de rastreio é mostrado no campo Caminho do Arquivo de Log.

No UNIX, abra o arquivo `odbc.ini`. O caminho para o arquivo de log de rastreio é indicado pela configuração `TraceFile`.

Configurando o Acesso às Origens de Dados ODBC

Para configurar o servidor federado para acessar as origens de dados ODBC, você deve fornecer ao servidor federado informações sobre as origens de dados e os objetos que deseja acessar.

Antes de Iniciar

- O driver ODBC deve ser instalado e configurado no servidor que age como o servidor federado.
- A federação deve ser instalada em um servidor que atua como o servidor federado.
- Verifique a configuração do servidor federado.
- Verifique o parâmetro federado para assegurar que a federação está ativada.
- Dependendo do seu cliente de origem de dados, você deve se certificar de que as seguintes variáveis sejam especificadas em ambas variáveis de ambiente e no arquivo `db2dj.ini`:

- As variáveis de ambiente do sistema necessárias por seu cliente de origem de dados.
- A variável DB2LIBPATH do DB2 Profile Registry (db2set).

Para determinar quais variáveis são necessárias, consulte a documentação fornecida por seu cliente de origem de dados.

Por exemplo, se precisar especificar bibliotecas de tempo de execução, pode ser necessário especificar as seguintes variáveis de ambiente:

- LIBPATH em sistemas operacionais AIX.
- PATH em sistemas operacionais Windows.
- LD_LIBRARY_PATH em sistemas operacionais Linux e Solaris.

Nota: Se você configurar variáveis de ambiente do sistema ou modificar o arquivo `db2dj.ini`, você deve parar e reiniciar o InfoSphere Federation Server.

Sobre Esta Tarefa

Restrições

- O wrapper ODBC não pode ser utilizado para acessar nenhuma origem de dados da família DB2. Utilize o wrapper DRDA para acessar as origens de dados da família DB2.
- O wrapper ODBC não suporta as seguintes funções e instruções:
 - Transações two-phase commit
 - Instruções LOCK TABLE em apelidos
 - Recursos reprovados no ODBC 3.x
 - Drivers X/Open ou SQL/CLI
 - Apelidos do Procedimento Armazenado
 - Execução de atomicidade no nível de instrução utilizando instruções de ponto de salvamento remotos
 - Cursores WITH HOLD
- Para as origens de dados que não suportam as operações de atualização e de exclusão posicionadas, as instruções UPDATE e DELETE posicionadas e certas instruções UPDATE e DELETE procuradas em um apelido falharão se um índice exclusivo em colunas não-anuláveis não existir no apelido ou na sua tabela remota correspondente. O erro SQL30090 com o código de razão 21 é retornado quando essas instruções falham.
- O wrapper ODBC não suporta as instruções INSERT, UPDATE ou DELETE sobre as origens de dados que restringem o número de instruções ativas para cada conexão. Consulte a documentação da origem de dados para determinar se ela restringe o número de instruções ativas para cada conexão. Uma das origens de dados ODBC à qual esta restrição se aplica é o IBM Red Brick Warehouse.
- O wrapper ODBC não suporta operações em tabelas que contêm colunas com tipos de dados que utilizam indicadores de tipo de dados de SQL específicos de driver. O tipo de operações que não são suportadas incluía as instruções CREATE NICKNAME e SELECT no modo de passagem. O wrapper ODBC suporta somente os indicadores de tipo de dados de SQL definidos pelo padrão ODBC no *Microsoft ODBC Programmer's Reference*.

As origens de dados acessadas por meio da API ODBC são referidas neste texto como origens de dados ODBC.

Dependendo de suas necessidades, você poderá acessar os dados Excel utilizando o wrapper ODBC, em vez de utilizar o wrapper do Excel. Para configurar o wrapper ODBC para acessar dados do Excel, consulte [Acessando dados do Excel utilizando o wrapper ODBC](#).

Recomendação: Para Microsoft SQL Server, você deveria utilizar o wrapper do Microsoft SQL Server para acessar origens de dados do Microsoft SQL Server em vez de utilizar o wrapper ODBC. O wrapper Microsoft SQL Server oferece melhor desempenho de consulta e mais funcionalidade para o Microsoft SQL Server. Para obter informações sobre como configurar o wrapper Microsoft SQL Server, consulte [Configurando acesso às origens de dados do Microsoft SQL Server](#).

Procedimento

1. Utilize um dos métodos a seguir para preparar o servidor federado e o banco de dados federado dependendo do sistema operacional:
 - Preparar o servidor federado para acessar origens de dados através de ODBC (Windows).
 - Preparar o servidor federado para acessar origens de dados através de ODBC (Linux, UNIX).
2. Registre o wrapper ODBC.
3. Registre as definições do servidor para uma origem de dados ODBC.
4. Crie um mapeamento de usuário para uma origem de dados ODBC.
5. Teste a conexão com o servidor da origem de dados ODBC.
6. Registre apelidos para as tabelas e visualizações da origem de dados ODBC.

Preparando o Servidor Federado para Acessar as Origens de Dados por meio do ODBC (Windows)

Nos servidores federados que executam Windows, o servidor federado deve conseguir acessar as origens de dados ODBC. Para preparar o servidor federado, é preciso verificar as configurações no DNS do Sistema ODBC.

Procedimento

Verifique se o driver ODBC 3.x foi instalado e configurado no servidor federado. O nome do nó para a origem de dados ODBC deve ser definido no DSN do Sistema. Consulte a documentação do driver ODBC para obter os procedimentos de instalação e configuração.

Se você utilizou a janela do Microsoft ODBC Data Source Administrator para configurar o DSN, é possível verificar essa configuração no Painel de Controle. Certifique-se de que a origem de dados ODBC esteja registrada como um DSN do Sistema. Caso contrário, o servidor federado poderá não conseguir localizar o DSN.

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar o wrapper.

Preparando o Servidor Federado para Acessar Origens de Dados por meio do ODBC (Linux, UNIX)

Nos servidores federados que executam Linux ou UNIX, o servidor federado deve conseguir acessar as origens de dados ODBC. Para preparar o servidor federado, é preciso verificar as configurações no arquivo `odbc.ini`, criar links simbólicos e testar a conexão com as origens de dados ODBC.

Antes de Iniciar

Para sistema operacional Linux para System z, é necessário configurar o parâmetro **EnableDescribeParam** para 1 no arquivo de configuração `odbc.ini` do DSN quando você utiliza o wrapper ODBC para conectar-se a origens de dados do Sybase ou Oracle com os drivers de DataDirect Sybase Wire Protocol, Oracle Wire Protocol, ou Oracle Client:

```
EnableDescribeParam=1
```

Se o parâmetro **EnableDescribeParam** não for configurado para 1, a mensagem de erro 1822N é lançada para ligar variáveis de host non-char e varchar quando o objeto ODBC SERVER estiver configurado para o modo PUSHDOWN.

Procedimento

1. Verifique se o arquivo `odbc.ini` foi atualizado no servidor federado. Se o arquivo não existir, você poderá criá-lo em um editor de texto. Consulte a documentação do fornecedor do cliente ODBC para obter informações sobre o arquivo `odbc.ini`.
2. Configure o cliente ODBC.
Consulte a documentação do fornecedor de cliente ODBC para obter instruções sobre como configurar o cliente ODBC.
3. Se o cliente for DataDirect ODBC ou RedBrick, verifique se os links simbólicos adequados foram criados. Nos seguintes links simbólicos, `ODBC_CLIENT_DIR` é o diretório em que o cliente ODBC está instalado.

Sistema operacional do servidor federado	Etapa
Linux	<p>Crie os seguintes links simbólicos:</p> <pre>In -s \$ODBC_CLIENT_DIR/./locale /usr/local/locale In -s \$ODBC_CLIENT_DIR/libodbcinst.so /usr/lib/libodbcinst.so</pre> <p>Se estiver usando o driver DataDirect Technologies Connect for ODBC 4.2, também é preciso criar o seguinte link simbólico:</p> <pre>In -s \$ODBC_CLIENT_DIR/libivicu19.so /usr/lib/libivicu19.so</pre>
Solaris	<p>Crie o seguinte link simbólico:</p> <pre>In -s \$ODBC_CLIENT_DIR/./locale \$HOME/sql1lib/locale</pre> <p><code>\$HOME</code> é o diretório inicial do proprietário da instância do DB2.</p>

4. Se o cliente for DataDirect ODBC, será possível testar a conexão do servidor federado com a a origem de dados utilizando a ferramenta `demoodbc` do DataDirect Connect ODBC.
 - a. Execute o script `/opt/odbc/odbc.sh`. Este script configura diversas variáveis de ambiente específicas do sistema operacional.
 - b. Teste a conexão com a origem de dados ODBC utilizando a ferramenta `demoodbc` do DataDirect Connect ODBC. A ferramenta `demoodbc` está localizada no subdiretório `/demo` das bibliotecas do Connect ODBC.

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar o wrapper.

Registrando o Wrapper ODBC

Você deve registrar um wrapper para acessar as origens de dados ODBC. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Procedimento

Utilize um dos seguintes métodos:

Método	Descrição
Emita a instrução CREATE WRAPPER a partir da linha de comandos e especifique o nome padrão para o wrapper ODBC.	<p>Por exemplo:</p> <pre>CREATE WRAPPER ODBC;</pre> <p>Lembre-se: Ao registrar o wrapper utilizando o nome padrão, o servidor federado utiliza automaticamente a biblioteca do wrapper ODBC apropriada para o sistema operacional no qual o servidor federado está em execução.</p> <p>É preciso especificar a opção do wrapper MODULE nos servidores federados que executam Linux ou UNIX. A opção do wrapper MODULE especifica o caminho completo da biblioteca que contém o Gerenciador de Driver ODBC.</p> <p>Para o AIX e o Solaris, se você estiver configurando o acesso para as origens de dados do IBM InfoSphere Classic Federation Server para z/OS e usar um cliente IBM InfoSphere Classic Federation Server para z/OS de 32 bits, você deverá especificar as opções DB2_FENCED e DB2_SOURCE_CLIENT_MODE, conforme mostra o exemplo a seguir.</p> <pre>CREATE WRAPPER ODBC LIBRARY 'libdb2rcodbc.a' OPTIONS (DB2_FENCED 'Y', DB2_SOURCE_CLIENT_MODE '32BIT', MODULE '/opt/IBM/DB2IIClassic82/cli/lib/cacsq1cli.so')</pre> <p>O DB2_SOURCE_CLIENT_MODE é suportado apenas no AIX e no Solaris.</p>
Emita a instrução CREATE WRAPPER a partir da linha de comandos e especifique um nome alternativo para o wrapper ODBC.	<p>Se o nome padrão do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, você poderá substituir o nome padrão do wrapper por um de sua escolha. Se você não utilizar o nome padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.</p> <p>Por exemplo, para registrar um wrapper com o nome odb_wrapper em um servidor federado que utiliza AIX, emita a seguinte instrução:</p> <pre>CREATE WRAPPER odb_wrapper LIBRARY 'libdb2rcodbc.a';</pre> <p>O arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado.</p>

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar as definições do servidor.

Arquivos de Biblioteca do Wrapper ODBC

Os arquivos da biblioteca do wrapper ODBC são incluídos no servidor federado ao instalar o wrapper.

Ao instalar o wrapper ODBC, três arquivos de biblioteca são incluídos no caminho do diretório padrão. Por exemplo, se o servidor federado estiver em execução no AIX, os arquivos de biblioteca do wrapper incluídos no caminho do diretório são `libdb2rcodbc.a`, `libdb2rcodbcF.a` e `libdb2rcodbcU.a`. O arquivo da biblioteca do wrapper padrão é `libdb2rcodbc.a`. Os outros arquivos da biblioteca do wrapper são usados internamente pelo wrapper ODBC.

Se não utilizar o nome padrão do wrapper ao registrá-lo, você deverá incluir o parâmetro `LIBRARY` na instrução `CREATE WRAPPER` e especificar o nome do arquivo da biblioteca padrão do wrapper.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 27. Locais de Biblioteca e Nomes de Arquivos do Cliente ODBC

Sistema Operacional	Caminho do Diretório	Arquivo da Biblioteca do Wrapper
AIX	<code>/usr/opt/install_path/lib32/</code> <code>/usr/opt/install_path/lib64/</code>	<code>libdb2rcodbc.a</code>
Linux	<code>/opt/IBM/db2/install_path/lib32</code> <code>/opt/IBM/db2/install_path/lib64</code>	<code>libdb2rcodbc.so</code>
Solaris	<code>/opt/IBM/db2/install_path/lib32</code> <code>/opt/IBM/db2/install_path/lib64</code>	<code>libdb2rcodbc.so</code>
Windows	<code>%DB2PATH%\bin</code>	<code>db2rcodbc.dll</code>

`install_path` é o caminho do diretório onde a federação é instalada no UNIX ou Linux.

Instrução CREATE WRAPPER - Exemplos para o wrapper ODBC

Utilize a instrução `CREATE WRAPPER` para registrar o wrapper ODBC. Este tópico fornece exemplos para Linux, UNIX e Windows.

Nos seguintes exemplos, `odbc_wrapper` é o nome atribuído ao wrapper que você está registrando no banco de dados federado.

Servidor Federado Linux e Solaris

O exemplo a seguir mostra como registrar um wrapper com o nome padrão em um servidor federado que executa Linux ou Solaris:

```
CREATE WRAPPER odbc OPTIONS (MODULE  
'/opt/lib/odbc.so');
```

É preciso especificar a opção do wrapper `MODULE` nos servidores federados que executam Linux ou UNIX. A opção do wrapper `MODULE` especifica o caminho completo da biblioteca que contém o Gerenciador de Driver ODBC.

O exemplo a seguir mostra como registrar um wrapper com um nome alternativo em um servidor federado que executa Linux ou Solaris:

```
CREATE WRAPPER odbc_wrapper
LIBRARY 'libdb2rcodbc.so'
      OPTIONS (MODULE '/opt/lib/odbc.so');
```

Servidor Federado AIX

O exemplo a seguir mostra como registrar um wrapper com o nome padrão em um servidor federado que executa AIX:

```
CREATE WRAPPER odbc
      OPTIONS (MODULE '/usr/lib/odbc.a');
```

É preciso especificar a opção do wrapper MODULE em servidores federados que executam UNIX. A opção do wrapper MODULE especifica o caminho completo da biblioteca que contém o Gerenciador de Driver ODBC.

O exemplo a seguir mostra como registrar um wrapper com um nome alternativo em um servidor federado que executa AIX:

```
CREATE WRAPPER odbc_wrapper LIBRARY
'libdb2rcodbc.a'
      OPTIONS (MODULE '/usr/lib/odbc.a');
```

IBM InfoSphere Classic Federation Server para z/OS (AIX)

O exemplo a seguir mostra como registrar um wrapper ODBC emitindo a instrução CREATE WRAPPER em um sistema operacional AIX. Para o AIX e o Solaris de 64 bits, se você usar um cliente IBM InfoSphere Classic Federation Server para z/OS, as opções DB2_FENCED e DB2_SOURCE_CLIENT_MODE deverão ser especificadas da maneira mostrada no exemplo a seguir.

```
CREATE
WRAPPER odbc
      OPTIONS (DB2_FENCED 'Y', DB2_SOURCE_CLIENT_MODE '32BIT',
      MODULE '/opt/IBM/DB2IIClassic82/cli/lib/cacsqlcli.so');
```

O DB2_SOURCE_CLIENT_MODE é suportado apenas no AIX e no Solaris.

É preciso especificar a opção do wrapper MODULE em servidores federados que executam UNIX. A opção do wrapper MODULE especifica o caminho completo da biblioteca que contém o Gerenciador de Driver ODBC.

O exemplo a seguir mostra como registrar um wrapper para acessar origens de dados do IBM InfoSphere Classic Federation Server para z/OS com um nome alternativo:

```
CREATE WRAPPER
odbc_wrapper LIBRARY 'libdb2rcodbc.a'
      OPTIONS (DB2_FENCED 'Y', DB2_SOURCE_CLIENT_MODE '32BIT',
      MODULE '/opt/IBM/DB2IIClassic82/cli/lib/cacsqlcli.so');
```

Servidor Federado Windows

O exemplo a seguir mostra como registrar um wrapper com o nome padrão em um servidor federado que executa Windows:

```
CREATE WRAPPER odbc;
```

O exemplo a seguir mostra como registrar um wrapper com um nome alternativo em um servidor federado que executa Windows:

```
CREATE WRAPPER odbc_wrapper LIBRARY  
'db2rcodbc.d11';
```

Registrando as Definições de Servidor para uma Origem de Dados ODBC

Você deverá registrar cada servidor ODBC que deseja acessar no banco de dados federado.

Procedimento

1. Emita a instrução CREATE SERVER a partir da linha de comandos e especifique as opções do servidor necessárias.

Por exemplo:

```
CREATE SERVER server_definition_name TYPE data_source_type  
  VERSION version_number WRAPPER wrapper_name  
  OPTIONS (NODE 'node_name', CODEPAGE 'codepage_name',  
  DBNAME 'database_name');
```

Embora NODE, CODEPAGE e DBNAME sejam opções do servidor na instrução CREATE SERVER, são necessárias para origens de dados ODBC:

- A opção do servidor NODE deve ser configurada para a origem de dados especificada na palavra-chave DATASOURCE no arquivo cac.ini.

Exemplo

Se o arquivo cac.ini especifica DATASOURCE = CACSAMP tcp/150.45.37.49/5000, então a opção NODE deve ser configurada para CACSAMP.

- A opção do servidor CODEPAGE deve ser configurada para o número de página de códigos da página de códigos do cliente especificado no arquivo cac.ini. A página de códigos do cliente é a página de códigos de origem de dados.

Exemplo

Se o arquivo cac.ini especifica CLIENT CODEPAGE = IBM-850, então a opção CODEPAGE deve ser configurada para 850.

- A opção do servidor DBNAME deve ser configurada para o nome do banco de dados da origem de dados que você deseja acessar.

Exemplo

Se o nome da origem de dados ODBC for venice, então a opção DBNAME deve estar configurada para *venice*.

2. Após a criação da definição do servidor, utilize a instrução ALTER SERVER para incluir ou descartar opções do servidor.

O que Fazer Depois

Após concluir esta tarefa, você pode criar um mapeamento de usuário.

Instrução CREATE SERVER - Exemplos do Wrapper ODBC

Utilize a instrução CREATE SERVER para registrar definições de servidor para o wrapper ODBC. Este tópico fornece um exemplo completo com os parâmetros requeridos e um exemplo com opções de servidor adicionais.

O exemplo a seguir mostra como registrar uma definição de servidor para uma origem de dados MySQL emitindo a instrução CREATE SERVER:

```
CREATE SERVER mysql_server
TYPE mysql
    VERSION 4.0 WRAPPER
wrapper_name
    OPTIONS (NODE 'odbc_node', DBNAME
'venice')
```

mysql_server

Um nome que você atribui ao servidor de origem de dados ODBC. Nomes de definição de servidor duplicados não são permitidos.

TYPE *mysql*

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso. Este parâmetro é opcional.

VERSION 4.0

A versão da origem de dados ODBC que deseja acessar. Este parâmetro é opcional.

WRAPPER *wrapper_name*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

NODE '*odbc_node*'

O nome do nó (o nome DSN do sistema) designado para a origem de dados ODBC quando o DSN foi definido. Em servidores federados que executam Windows, este valor deve ser o nome de um DSN do sistema na janela do Administrador da Origem de Dados ODBC. Em servidores federados que executam UNIX, o nome do nó é o DSN definido no arquivo de configuração de ODBC. O arquivo de configuração de ODBC é usualmente chamado de *odbc.ini*.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é necessário para origens de dados ODBC.

DBNAME '*venice*'

Opcional. O nome da origem de dados ODBC que deseja acessar.

Opções do Servidor

Ao criar a definição do servidor, você pode especificar opções de servidor adicionais na instrução CREATE SERVER. As opções do servidor podem ser gerais ou específicas do ODBC.

Algumas origens de dados ODBC (por exemplo, MySQL) não podem processar aspas em torno de nomes de tabelas e de colunas nas instruções SQL. Para acessar essas origens de dados, é preciso incluir as seguintes opções de servidor na instrução CREATE SERVER:

- DB2_TABLE_QUOTE_CHAR '``'
- DB2_ID_QUOTE_CHAR '``'
- DB2_AUTHID_QUOTE_CHAR '``'

O caractere ` é o delimitador para identificadores, como, por exemplo, nomes de esquemas, nomes de tabelas e nomes de colunas.

Por exemplo:

```
CREATE SERVER mysql_server
TYPE mysql
    VERSION 4.0 WRAPPER
wrapper_name
```

```
OPTIONS (NODE 'mysql_node',
DB2_TABLE_QUOTE_CHAR '~',
DB2_ID_QUOTE_CHAR '~',
DB2_AUTHID_QUOTE_CHAR '~')
```

Criando um Mapeamento de Usuário para uma Origem de Dados ODBC

Quando você tenta acessar um servidor ODBC, o servidor federado estabelece uma conexão com o servidor ODBC utilizando um ID de usuário e uma senha válidos para a origem de dados. Para origens de dados que exigem um mapeamento de usuário, você deve definir uma associação (um mapeamento de usuário) entre o ID de usuário e a senha de cada servidor federado e o ID de usuário e a senha correspondentes da origem de dados.

Sobre Esta Tarefa

Crie um mapeamento de usuário para cada ID de usuário que acessará o sistema federado para enviar pedidos distribuídos à origem de dados ODBC.

Procedimento

Emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR local_userID SERVER server_definition_name
OPTIONS (REMOTE_AUTHID 'remote_userID', REMOTE_PASSWORD 'remote_password');
```

Embora REMOTE_AUTHID e REMOTE_PASSWORD sejam opções de mapeamento de usuário na instrução CREATE USER MAPPING, essas opções são necessárias para acessar origens de dados ODBC:

O mapeamento de usuário deveria mapear o DB2 auth ID para ID do usuário e senha especificados no arquivo cac.ini.

Exemplo

Se o arquivo cac.ini especifica USERID = *MY_USERID* e USERPASSWORD = *MY_PASSWORD*, então as opções da instrução CREATE USER MAPPING devem ser especificadas como segue:

```
REMOTE_AUTHID = MY_USERID
REMOTE_PASSWORD = MY_PASSWORD
```

O que Fazer Depois

Após concluir esta tarefa, será possível testar a conexão com a origem de dados ODBC.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper ODBC

Utilize a instrução CREATE USER MAPPING para mapear um ID de usuário do servidor federado para um ID de usuário e senha da origem de dados ODBC. Este tópico fornece um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o USER de registro especial do DB2 com a instrução CREATE USER MAPPING.

O exemplo a seguir mostra como mapear um ID de autorização federado para um ID de usuário e uma senha da origem de dados ODBC:


```
CREATE USER MAPPING FOR
arturo SERVER mysql_server
      OPTIONS (REMOTE_AUTHID 'art',
REMOTE_PASSWORD 'red4blue')
```

arturo Especifica o ID de autorização local que você está mapeando para o ID de usuário e senha remotos, definidos na origem de dados ODBC.

mysql_server

Especifica o nome da definição do servidor definido na instrução CREATE SERVER para a origem de dados ODBC.

'art' Especifica o ID de usuário remoto para o qual você está mapeando *arturo*. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_ID para 'U' ou 'L' na instrução CREATE SERVER.

'red4blue'

Especifica a senha remota associada a *'art'*. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

USER de Registro Especial do DB2

Você pode utilizar o USER de registro especial do DB2 para mapear o ID de autorização da pessoa que está emitindo a instrução CREATE USER MAPPING para o ID de autorização da origem de dados especificado na opção de usuário REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER SERVER
mysql_server
      OPTIONS (REMOTE_AUTHID 'art',
REMOTE_PASSWORD 'red4blue');
```

Testando a Conexão com o Servidor de Origem de Dados ODBC

Teste a conexão com o servidor da origem de dados ODBC para determinar se o servidor federado está configurado adequadamente para acessar as origens de dados ODBC.

Sobre Esta Tarefa

Você pode testar a conexão com o servidor do origem de dados ODBC utilizando a definição do servidor e os mapeamentos de usuário que definiu.

Procedimento

Abra um sessão de passagem e emita uma instrução SELECT nas tabelas do sistema da origem de dados ODBC. Se a instrução SELECT retornar uma contagem, a definição do servidor e o mapeamento de usuário estão configurados apropriadamente.

```
SET PASSTHRU server_definition_name
SELECT count(*) FROM schema_name.table_name
SET PASSTHRU RESET
```

Se a instrução SELECT retornar um erro, você deverá resolver os problemas dos erros de conexão.

O que Fazer Depois

Após concluir esta tarefa, será possível registrar apelidos para as tabelas e visualizações da origem de dados ODBC.

Resolução de Problemas de Erros de Conexão da Origem de Dados

Uma conexão de teste com o servidor de origem de dados pode retornar um erro por diversas razões. Há ações que você pode executar para determinar a causa de um erro.

Sintoma

É retornado um erro quando você tentar se conectar à origem de dados.

Causa

Há diversas causas possíveis para um problema de conexão.

Solução do Problema

Para resolver problemas de erros de conexão com a origem de dados, verifique os seguintes itens:

- Verifique se a origem de dados está disponível.
- Se aplicável, certifique-se de que o servidor de origem de dados esteja configurado para conexões de chegada.
- Certifique-se de suas configurações de mapeamento de usuário para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com a origem de dados. Altere o mapeamento de usuário ou crie outro, conforme necessário.
- Se aplicável, certifique-se de que o software do cliente da origem de dados no servidor federado esteja instalado e configurado corretamente para conexão com a origem de dados.
- Para origens de dados ODBC, certifique-se de que o driver ODBC no servidor federado esteja instalado e configurado corretamente para conexão com o servidor de origem de dados ODBC. Em servidores federados que executam Windows, utilize a ferramenta Administrador de Origem de Dados ODBC para verificar o driver. Em servidores de dados que executam UNIX, consulte a documentação do fornecedor do cliente ODBC.
- Verifique se as configurações para as variáveis configuradas no servidor federado estão corretas para a origem de dados. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis no arquivo db2dj.ini e as variáveis do DB2 Profile Registry (db2set).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e a crie novamente.

Registrando Apelidos para Tabelas e Visualizações de Origens de Dados ODBC

Para cada definição do servidor ODBC registrada, você deverá registrar um apelido para cada tabela ou visualização que deseja acessar. Utilize esses apelidos em vez dos nomes dos objetos de origem de dados quando consultar as origens de dados ODBC.

Antes de Iniciar

Atualize as estatísticas na origem de dados ODBC antes de registrar um apelido. O banco de dados federado confia nas estatísticas do catálogo da origem de dados para otimizar o processamento da consulta. Utilize o comando da origem de dados equivalente ao comando do DB2 **RUNSTATS** para atualizar as estatísticas da origem de dados.

Procedimento

Emita a instrução **CREATE NICKNAME** a partir da linha de comandos.

Por exemplo:

```
CREATE NICKNAME nickname
FOR server_definition_name."remote_schema"."remote.table" ;
```

Os apelidos podem ter até 128 caracteres de comprimento.

Ao criar um apelido, o servidor federado consulta o catálogo de origem de dados utilizando o apelido. Esta consulta testa a conexão com a tabela ou visualização da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

O que Fazer Depois

Repita esta etapa para cada tabela ODBC ou visualização para as quais deseja criar um apelido.

Instrução **CREATE NICKNAME** - Exemplos para o Wrapper ODBC

Utilize a instrução **CREATE NICKNAME** para registrar um apelido para uma tabela ou visualização do ODBC que deseja acessar. Este tópico fornece um exemplo completo com os parâmetros necessários.

O exemplo a seguir mostra como registrar um apelido para uma tabela ou visualização do ODBC utilizando a instrução **CREATE NICKNAME**.

```
CREATE NICKNAME
cust_europe FOR mysql_server."vinnie"."italy"
```

cust_europe

Um apelido exclusivo que é utilizado para identificar a tabela ou visualização do ODBC. O apelido deve ser exclusivo dentro do esquema.

Importante: O apelido é um nome de duas partes -o esquema e o apelido. Se você omitir o esquema ao registrar o apelido, o esquema do apelido será o ID de autorização do usuário que registrar o apelido.

Se a origem de dados ODBC não suportar esquemas, omita o esquema da instrução **CREATE NICKNAME**.

```
mysql_server."vinnie"."italy"
```

Um identificador de três partes para o objeto remoto:

- *mysql_server* é o nome da definição do servidor atribuído ao servidor da origem de dados ODBC na instrução CREATE SERVER.
- *vinnie* é o ID de usuário do proprietário ao qual a tabela ou visualização pertence.
- *italy* é o nome da tabela ou visualização remota que você deseja acessar.

O servidor federado fecha os nomes dos esquemas e tabelas do ODBC para maiúsculas, a menos que você coloque os nomes entre aspas.

Otimizando o Desempenho do Wrapper ODBC com o Utilitário de Ajuste do ODBC (db2fedsrvrfg)

É possível otimizar o desempenho do wrapper ODBC com o utilitário de ajuste de ODBC, o db2fedsrvrfg. Esse utilitário executa um conjunto de consultas predefinidas sobre a origem de dados e testa os resultados quanto à exatidão. O utilitário cria um conjunto de instruções ALTER SERVER que você pode executar sobre o servidor para configurar as opções do servidor a fim de obter o desempenho ideal.

Sobre Esta Tarefa

Os seguintes itens devem ser configurados no seu servidor federado:

- Federação
- O wrapper ODBC
- O software de cliente ODBC
- Variáveis para o ambiente do sistema. As variáveis ODBCINI e LIBPATH podem ser necessárias.

Para obter informações sobre instalação, configuração e requisitos de ODBC, consulte a documentação que é fornecida com cada um desses produtos.

Procedimento

1. If your ODBC server has not been defined, use the DB2 Command Line Processor to connect to the federated server and create an ODBC wrapper and server.
2. Opcional: Se as tabelas de teste já existirem na origem de dados de ODBC, conecte-as à origem de dados ODBC e descarte-as.
3. Execute o utilitário de ajuste de ODBC a partir da linha de comandos do DB2 com as opções que deseja utilizar. O utilitário pode levar algum tempo para concluir.
4. Verifique se a execução do utilitário foi bem-sucedida. Se a execução do utilitário foi bem-sucedida, a seguinte mensagem é vista na janela de comandos ou no arquivo de saída que você especificou:

```
ALTER SERVER "DS1" OPTIONS (ADD
option1, 'value1')
ALTER SERVER "DS1" OPTIONS (ADD option2,
'value2')
ALTER SERVER "DS1" OPTIONS (ADD option3,
'value3')
.....
```

A conclusão do comando db2fedsrvrfg foi bem-sucedida.

Se a execução do comando foi bem-sucedida, uma mensagem de erro que indica o motivo do erro é vista. Corrija o problema e execute o comando novamente.

- É preciso criar as tabelas de teste manualmente nas seguintes circunstâncias:
- Se desejar utilizar nomes de tabelas que são diferentes do padrão
 - Se a origem de dados que você estiver acessando por meio de ODBC for de leitura
 - Se o utilitário de ajuste de ODBC não conseguir criar as tabelas de teste que são exigidas pelo utilitário
5. Conecte ao servidor federado onde a origem de dados for definida.
 6. Use o utilitário db2look para salvar as configurações de servidor existentes antes de executar o arquivo que é criado pelo utilitário de ajuste de ODBC. Consulte a documentação do utilitário db2look para obter informações sobre como salvar as configurações de servidor existentes.
 7. Opcional: Se o servidor ODBC for definido, será possível conectar ao servidor federado e descartar as opções de servidor. O utilitário cria instruções ALTER SERVER no formato descrito na etapa 4. Se essas opções de servidor já tiverem sido incluídas, as instruções ALTER SERVER falharão.
 8. Use o seguinte comando para executar as instruções ALTER SERVER geradas pelo utilitário sobre o banco de dados federado. As instruções ALTER SERVER criadas pelo utilitário de ajuste de ODBC estão contidas no arquivo db2fedsvrcfg.sql.


```
db2 -tvf db2fedsvrcfg.sql
```
 9. Verifique os resultados das instruções ALTER SERVER. Se alguma das instruções falhar, será possível modificar as instruções no arquivo db2fedsvrcfg.sql e executar as instruções novamente até que elas sejam bem-sucedidas.
 10. Após concluir o ajuste do servidor com o utilitário de ajuste de ODBC, configure a opção de servidor PUSHDOWN como 'Y' para concluir o processo de otimização.

Sintaxe do Comando db2fedsvrcfg - Utilitário de Ajuste de ODBC

Utilize o comando **db2fedsvrcfg** para aprimorar o desempenho do wrapper ODBC.

Sintaxe

```
db2fedsvrcfg -s serverName [-m odbcDriverManagerLibrary] -dsn odbcDSNname
[-dbname dsDBname] [-u userid] [-p password] [-noprep] [-prefix tableNamePrefix] [-suffix
tableNameSuffix] [-dscp codePage] [-v] [-o outputFile] [-h]
```

Parâmetros

Utilize o comando **db2fedsvrcfg32** ao utilizar os drivers ODBC de 32 bits no AIX ou no Solaris. Caso contrário, use o comando **db2fedsvrcfg**.

-dbname *dsDBname*

O nome do banco de dados da origem de dados.

-dscp *codePage*

O identificador da página de códigos para a origem de dados. Se esta opção não for especificada, o utilitário usa a página de códigos do ambiente do usuário. Este parâmetro é opcional.

-dsn *odbcDSNname*

O DSN (nome da origem de dados) do sistema para a origem de dados.

-h Faz com que a ajuda detalhada seja exibida. Este parâmetro é opcional.

-m odbcDriverManagerLibrary

O nome totalmente qualificado do arquivo da biblioteca do gerenciador de driver ODBC. O nome do arquivo de biblioteca do gerenciador de driver ODBC é opcional para oWindows.

-noprep

Evita que as tabelas de teste sejam criadas na origem de dados antes do teste. Este parâmetro é opcional.

-o outputFile

O nome totalmente qualificado do arquivo para o arquivo de saída do utilitário de ajuste de ODBC. O arquivo de saída contém as instruções ALTER SERVER que são usadas para ajustar o desempenho do wrapper ODBC. Este parâmetro é opcional. Se este parâmetro não for especificado, a saída será exibida na janela de comandos.

-p password

A senha para conectar à origem de dados. Este parâmetro é opcional.

-prefix tableNamePrefix

O prefixo dos nomes de tabela da origem de dados ODBC que o utilitário usa para a análise. Se um prefixo não for especificado, o prefixo padrão, IITEST, será usado. Este parâmetro é opcional.

-s serverName

O nome do servidor federado.

-suffix tableNameSuffix

O sufixo dos nomes de tabela da origem de dados ODBC que o utilitário usa para a análise. Se um sufixo não for especificado, uma cadeia vazia será usada.

-u userid

O nome do usuário para conectar à origem de dados. Este parâmetro é opcional.

-v Especifica que a saída do utilitário é detalhada. Este parâmetro é opcional.

Exemplo

O exemplo a seguir mostra o comando que é executado sobre o armazém de dados, a origem de dados ODBC. Neste exemplo, as tabelas de teste são nomeadas ABC n XYZ, em que n é um número de 1 a 7.

db2fedsvr cfg

```
-s DS1 -m "/usr/lib/odbc.a"  
-dsn datastore -dbname db1  
-u authid -p password  
-noprep  
-prefix ABC -suffix XYZ  
-o "/home/user1/db2fedsvr cfg.sql"
```

Definições da tabela de teste para o utilitário de ajuste de ODBC (db2fedsvr cfg)

Em alguns casos, é preciso criar as tabelas de teste para o utilitário de ajuste de ODBC manualmente. As definições da tabela de teste são descritas neste tópico.

É preciso criar as tabelas de teste para o utilitário de ajuste de ODBC nas seguintes circunstâncias:

- Se desejar utilizar nomes de tabelas que são diferentes do padrão
- Se a origem de dados que você estiver acessando por meio de ODBC for de leitura

- Se o utilitário de ajuste de ODBC não conseguir criar as tabelas de teste que são exigidas pelo utilitário

A seguinte definição de tabela aplica-se a todas as sete tabelas de teste que são necessárias (IITEST1 por meio de IITEST7). O prefixo padrão do nome da tabela é IITEST, e o sufixo padrão é uma cadeia vazia. Se você especificar um prefixo e um sufixo diferentes, é preciso especificar as opções -prefix e -suffix ao executar o utilitário de ajuste de ODBC.

Tabela 28. Definição da Tabela de Teste do Utilitário de Ajuste de ODBC para a Tabela IITEST1

Nome da Coluna	Tipo de dados do SQL	Identificador de tipo de dados do SQL	Comprimento
IT1C1	número inteiro	SQL_INTEGER	
IT1C2	número inteiro	SQL_INTEGER	
IT1C3	char(1)	SQL_CHAR	1
IT1C4	char(3)	SQL_CHAR	3
IT1C5	char(10)	SQL_CHAR	10
IT1C6	varchar(10)	SQL_VARCHAR	10
IT1C7	char(100)	SQL_CHAR	100

Tabela 29. Definição da Tabela de Teste do Utilitário de Ajuste de ODBC para a Tabela IITEST2

Nome da Coluna	Tipo de dados do SQL	Identificador de tipo de dados do SQL	Comprimento
IT2C1	número inteiro	SQL_INTEGER	
IT2C2	número inteiro	SQL_INTEGER	
IT2C3	char(30)	SQL_CHAR	30

Tabela 30. Definição da Tabela de Teste do Utilitário de Ajuste de ODBC para as Tabelas IITEST3 a IITEST7

Número de colunas em cada tabela	Nome da Coluna	Tipo de dados do SQL	Identificador de tipo de dados do SQL	Comprimento
66	ITxCn	char(100)	SQL_CHAR	100

x O número correspondente à tabela que está sendo definida.

n O número da coluna.

Por exemplo, IT3C1 é o nome de coluna para a primeira coluna na tabela IITEST3.

Acessando Dados do Excel Utilizando o Wrapper ODBC

Você pode acessar as pastas de trabalho do Microsoft Excel com o wrapper ODBC utilizando o driver ODBC do Excel.

Antes de Iniciar

- O driver ODBC do Excel deve estar no servidor federado.
- O servidor federado deve conseguir abrir e ler as planilhas na pasta de trabalho do Excel para recuperar os dados. Portanto, as pastas de trabalho do Excel devem estar no mesmo computador que o servidor federado ou em uma unidade de rede mapeada acessível.
- Formate as colunas de acordo com o tipo de dados que se espera que a coluna tenha.

- Os dados inseridos nas colunas devem estar de acordo com o tipo de formato que é especificado para a coluna.
- Se as primeiras oito linhas da planilha não possuírem dados, verifique se elas estão vazias. Para certificar-se de que uma célula esteja vazia, abra a planilha no Microsoft Excel e selecione **Editar > Limpar tudo**.
- Certifique-se de que os dados inseridos nas colunas na planilha estejam de acordo com o tipo especificado.

Sobre Esta Tarefa

O aplicativo do Excel não precisa ser instalado no servidor federado. O driver ODBC do Excel é instalado automaticamente com o Windows.

Com o wrapper ODBC e o driver ODBC do Excel, é possível acessar dados de qualquer planilha dentro de uma pasta de trabalho. O driver ODBC do Excel interpreta uma pasta de trabalho como um banco de dados e interpreta cada planilha dentro da pasta de trabalho como uma tabela.

O driver ODBC do Excel suporta versões anteriores das pastas de trabalho do Excel mesmo se a versão do aplicativo Excel que produziu as pastas de trabalho não seja mais suportada. Por exemplo, a Microsoft não mais suporta planilhas criadas no Excel Versão 4.0, mas o driver suporta planilhas do Excel que foram criadas nessa versão.

Restrições

- O wrapper ODBC não pode acessar uma planilha quando a pasta de trabalho já estiver aberta por um usuário ou por um aplicativo no modo leitura/gravação. Entretanto, se o wrapper ODBC abrir a pasta de trabalho antes de um usuário ou aplicativo abri-la, o usuário ou aplicativo poderá abri-la no modo leitura/gravação.
- O driver ODBC do Excel espera que a primeira linha que não está em branco contenha os rótulos das colunas da planilha. É preciso inserir uma linha dos rótulos da coluna na planilha se a planilha não possuir os rótulos.
- Como o driver ODBC do Excel está disponível apenas para sistemas operacionais Windows, você poderá utilizar o wrapper ODBC para acessar os dados Excel somente nos servidores federados que executam Windows.
- Você pode executar operações de inserção e de atualização em planilhas do Excel, mas não pode executar operações de exclusão. O driver ODBC do Excel não suporta operações de exclusão. Para excluir dados da planilha, é preciso abrir a planilha no Excel para fazer as alterações.

Procedimento

1. Certifique-se de que a pasta de trabalho do Excel que você deseja acessar esteja no servidor federado ou em uma unidade de rede mapeada acessível.
2. Se seus dados do Excel estão compartilhados através de uma rede Windows transicional que utiliza grupos de trabalho, é necessário configurar suas permissões de acesso na origem de dados do Excel.
3. Se necessário, altere o layout dos dados nas planilhas do Excel para seguir os requisitos do driver ODBC do Excel. Repita esta etapa para cada planilha ou intervalo nomeado que desejar acessar.
4. Se necessário, crie qualquer intervalo nomeado que desejar acessar.
5. Crie um DSN do sistema para a pasta de trabalho que desejar acessar. Você pode utilizar o ODBC Data Source Administrator para configurar o DSN do

sistema. O nome especificado quando você criou o DSN do sistema é atribuído como o valor para a opção NODE na instrução CREATE SERVER.

Se sua origem de dados Excel estiver compartilhada através de uma rede Windows que utiliza grupos de trabalho, é necessário especificar o nome do banco de dados do sistema DSN com a seguinte sintaxe:

```
\\computer_Name\filename_Subdirectory
```

em que *computer_Name* é o nome do computador da origem de dados do Excel e *filename_Subdirectory* é o subdiretório e nome do arquivo do Excel.

Exemplo

Se o nome do computador da origem de dados do Excel for XLSQLS e o diretório de rede para o arquivo do Excel for E:\share\test.xls, então especifique o seguinte nome do banco de dados DSN:

```
\\XLSQLS\share\test.xls
```

em que o diretório-raiz do diretório de rede E: é substituído por \\ e o nome do computador XLSQLS.

6. Emita a instrução CREATE WRAPPER.
7. Especifique o local da pasta de trabalho registrando um objeto do servidor no catálogo do sistema do banco de dados federado. Para o wrapper ODBC, você precisa de um objeto do servidor para cada DSN. O DSN está associado à pasta de trabalho quando o driver ODBC do Excel é utilizado. O NODE *compounds_workbook_dsn* é o DSN de sistema que você criou. A opção NODE é necessária para que o wrapper ODBC acesse as planilhas do Excel.

Para especificar o local da pasta de trabalho, emita a instrução CREATE SERVER e use o DSN como o DSN do sistema para a opção NODE.

Por exemplo:

```
CREATE SERVER compounds_workbook
WRAPPER odbc
  OPTIONS (NODE
'compounds_workbook_dsn', PASSWORD
'n')
```

Repita esta etapa para cada pasta de trabalho que planejar acessar.

8. Emita a instrução CREATE NICKNAME para criar um apelido para a planilha que deseja acessar. A sintaxe é:

```
CREATE NICKNAME nickname FOR
server_name.remote_table
```

9. Se um intervalo nomeado foi criado para acessar os dados, especifique o nome do intervalo como a parte *remote_table* da instrução CREATE NICKNAME.

Por exemplo, se o nome do intervalo for *testing*, a instrução CREATE NICKNAME será:

```
CREATE NICKNAME compounds_nickname FOR
compounds_workbook.testing
```

Para acessar os dados em toda a planilha em vez de acessar em um intervalo, especifique o nome da planilha seguido pelo símbolo \$.

Por exemplo, se o nome da planilha for *Sheet1*, a instrução CREATE NICKNAME será:

```
CREATE NICKNAME compounds_nick FOR
compounds_workbook.Sheet1$
```

Configurando Permissões de Acesso para Dados do Excel em um Grupo de Trabalho ao Utilizar o Wrapper ODBC

É possível configurar suas permissões de acesso para acessar dados remotos do Excel que estão compartilhados através de uma rede Windows que utiliza grupos de trabalho.

Sobre Esta Tarefa

Todos os exemplos nesta tarefa são para o Windows XP Professional, consulte a documentação do sistema operacional onde está sua origem de dados para obter informações específicas sobre como configurar suas permissões de conta do usuário.

Procedimento

1. Configure o compartilhamento e o modelo de segurança no painel **Acesso à Rede: Compartilhamento e modelo de segurança para contas locais**. É possível selecionar um dos seguintes modelos de compartilhamento e modelo de segurança:

Somente Convidado – usuários locais autenticados como Convidados

Você conecta à origem de dados por meio somente da conta do usuário Convidado. Você recebe o nível de acesso que configurou para a conta de usuário Convidado e deve utilizar sua senha de usuário DB2.

Clássico – usuários locais autenticados como eles próprios.

É possível utilizar seu ID do usuário e senha DB2 para conectar à origem de dados ou utilizar a conta de usuário Convidado. Para o ID do usuário DB2, você recebe o nível de acesso para a origem de dados de acordo com o nível de acesso de seu ID do usuário DB2. Para a conta de usuário Convidado, você recebe o nível de acesso que configurou para a conta de usuário Convidado e deve utilizar sua senha de usuário DB2.

Exemplo

Para selecionar o modelo de segurança e compartilhamento **Clássico – usuários locais autenticados como eles próprios**, abra as **Ferramentas Administrativas** e navegue para **Política de Segurança Local > Políticas Locais > Opções de Segurança > Acesso à rede:Compartilhamento e modelo de segurança para contas locais**.

2. Crie a conta do usuário no painel **Acessar esse computador a partir das Propriedades de rede**:

Opção	Descrição
Para o modelo de segurança Somente convidado – usuários locais autenticados como Convidados	Especifique o grupo de trabalho Windows e crie a conta de usuário Convidado. Você deve especificar a senha utilizando sua senha do DB2.
Para o modelo de segurança Clássico – usuários locais autenticados como eles próprios	<ul style="list-style-type: none">• Especifique o grupo de trabalho Windows e crie a conta do usuário utilizando ID do usuário e senha do DB2.• Se deseja utilizar a conta de usuário Convidado, especifique o grupo de trabalho do Windows e crie a conta de usuário Convidado. Você deve especificar a senha para a conta de usuário Convidado utilizando sua senha do DB2.

Exemplo

Para criar a conta de usuário Convidado:

- a. Abra as **Ferramentas Administrativas** e navegue para **Política de Segurança Local > Políticas Locais > Designação de Direitos do Usuário > Acessar o computador a partir da rede**
 - b. Especifique o grupo de trabalho Windows no campo **Deste local**.
 - c. Especifique o ID do usuário DB2 no campo **Digitar os nomes de objeto para selecionar**.
 - d. Clique em **Verificar Nomes**.
 - e. No painel **Digitar Senha de Rede**, especifique o ID do usuário e senha do DB2.
3. Configure as permissões de compartilhamento no painel **Propriedades do Compartilhamento** configurando as permissões de **Alteração** e **Leitura** para **Permitir**.

Exemplo

Para configurar as permissões de compartilhamento do diretório de rede E:\share\test.xls:

- Clique com o botão direito do mouse na pasta do diretório E:\share e então clique em **Compartilhamento e Segurança**.
 - Na guia **Compartilhamento**, selecione **Compartilhar essa pasta** e clique em **Permissões**.
 - Selecione o usuário e, então, selecione **Permitir** para ambas as permissões, **Alteração** e **Leitura**.
4. Configurar as permissões de NTFS no painel **Propriedades** configurando as permissões **Leitura & Execução** e **Leitura** para **Permitir**.

Exemplo

Para configurar as permissões NTFS do arquivo E:\share\test.xls:

- Clique com o botão direito do mouse no arquivo test.xls e, então, clique em **Propriedades**.
- Na guia **Segurança**, selecione usuário e, então, selecione **Permitir** para ambas as permissões, **Leitura & Execução** e **Leitura**.

O que Fazer Depois

Complete o procedimento para acessar dados do Excel utilizando o wrapper ODBC.

Configurando Acesso ODBC para Origens de Dados do IBM InfoSphere Classic Federation Server para z/OS

O wrapper ODBC foi otimizado para acessar origens de dados do IBM InfoSphere Classic Federation Server para z/OS. O wrapper ODBC detecta o driver ODBC e configura automaticamente as opções de desempenho.

Antes de Iniciar

Os seguintes itens devem ser configurados no seu servidor federado:

- Servidor Federado
- Cliente do IBM InfoSphere Classic Federation Server para z/OS
- Banco de Dados Federado

- Variáveis no arquivo `db2dj.ini` de ambiente do sistema do registro de perfil DB2 (`db2set`):
 - Para sistemas Linux e UNIX, a variável `CAC_CONFIG` deve especificar o diretório do arquivo `cac.ini`, por exemplo:
`CAC_CONFIG=/home/db2inst1/cac.ini`
 - A variável de ambiente `DB2LIBPATH` pode ser necessária, dependendo dos diretórios de instalação da biblioteca do gerenciador de driver ODBC e da biblioteca de driver ODBC.
- Para os sistemas operacionais AIX e Solaris de 64 bits, se você usar um cliente IBM InfoSphere Classic Federation Server para z/OS de 32 bits, a opção do wrapper `DB2_FENCED` deverá ser configurada para `Y` e a opção do wrapper `DB2_SOURCE_CLIENT_MODE` deverá ser configurada para `32BIT`. A opção `DB2_SOURCE_CLIENT_MODE` apenas é suportada no AIX e no Solaris.

Para obter informações sobre instalação, configuração e requisitos de ODBC, consulte a documentação que é fornecida com cada um desses produtos.

Sobre Esta Tarefa

Restrições

O wrapper ODBC não suporta a seguinte instrução com origens de dados do IBM InfoSphere Classic Federation Server para z/OS:

- `CREATE TABLE`

Os seguintes tipos de dados não são suportados para uso com as origens de dados do IBM InfoSphere Classic Federation Server para z/OS:

- `BLOB`
- `CLOB`
- `DBCLOB`
- `CHAR FOR BIT DATA`
- `VARCHAR FOR BIT DATA`

Procedimento

1. Conclua uma das seguintes tarefas dependendo do seu sistema operacional:
 - No Windows, certifique-se de que o nome do nó para a origem de dados esteja definido como um DSN do sistema. Se foi utilizado o Microsoft ODBC Data Source Administrator para definir o DSN, é possível utilizar o **Painel de Controle** para verificar se ele está registrado como um sistema DSN.
 - No UNIX, configure o driver ODBC.
2. Pode ser necessário incluir o diretório de biblioteca do cliente na origem de dados na variável de ambiente `DB2LIBPATH`, para que as bibliotecas do cliente sejam carregadas corretamente. Execute o comando `db2set` para configurar a variável de ambiente `DB2LIBPATH` e especifique o diretório de instalação da biblioteca do cliente de origem de dados:
`db2set DB2LIBPATH="client_library_directory"`

Sendo que `client_library_directory` é o diretório da biblioteca do cliente da origem de dados.

Para versões de 64 bits, certifique-se de configurar a variável de ambiente `DB2LIBPATH` para o diretório `lib` de 64 bits.

Exemplos

Para o cliente do IBM InfoSphere Classic Federation Server para z/OS Versão 8.2:

```
db2set DB2LIBPATH="/opt/IBM/DB2IIClassic82/cli/lib"
```

Para o cliente do IBM InfoSphere Classic Federation Server para z/OS Versão 9.1:

```
db2set DB2LIBPATH="/opt/IBM/WSCClassic91/cli/lib"
```

Para o cliente do IBM InfoSphere Classic Federation Server para z/OS Versão 9.5:

```
db2set DB2LIBPATH="/opt/IBM/WSCClassic95/cli/lib"
```

3. Registre o wrapper ODBC.
4. Registre as definições do servidor para uma origem de dados ODBC.
5. Crie um mapeamento de usuário para uma origem de dados ODBC.
6. Teste a conexão com o servidor da origem de dados ODBC.
7. Registre apelidos para as tabelas e visualizações da origem de dados ODBC.

Configurando o Acesso ODBC às Origens de Dados Netezza

O wrapper ODBC é otimizado para acessar as origens de dados Netezza. O wrapper ODBC detecta o driver ODBC e configura automaticamente as opções de desempenho.

Antes de Iniciar

O driver ODBC do Netezza deverá ser configurado no servidor federado. O Netezza fornece um driver ODBC para os sistemas operacionais AIX, Linux e Windows.

Para obter os pré-requisitos que se aplicam a todas as origens de dados ODBC, consulte "Configurando o Acesso às Origens de Dados ODBC" na página 119.

Sobre Esta Tarefa

É possível usar a opção do servidor ENABLE_BULK_INSERT para usufruir das operações de inserção em massa para a origem de dados Netezza.

Restrições:

- As operações de inserção em massa não são suportadas em sessões de passagem.
- O Netezza não suporta todas as funções de federação. Para obter uma lista completa das restrições do Netezza, consulte a documentação do produto Netezza.

Restrições adicionais aplicam-se a todas as origens de dados ODBC. Consulte o "Configurando o Acesso às Origens de Dados ODBC" na página 119.

Procedimento

1. Utilize um dos métodos a seguir para preparar o servidor federado e o banco de dados federado dependendo do sistema operacional:
 - Preparar o servidor federado para acessar origens de dados através de ODBC (Windows).

- Preparar o servidor federado para acessar origens de dados através de ODBC (Linux, UNIX).
2. Pode ser necessário incluir o diretório de biblioteca do cliente na origem de dados na variável de ambiente DB2LIBPATH, para que as bibliotecas do cliente sejam carregadas corretamente. Execute o comando **db2set** para configurar a variável de ambiente DB2LIBPATH e especifique o diretório de instalação da biblioteca do cliente de origem de dados:


```
db2set DB2LIBPATH="client_library_directory"
```

Sendo que *client_library_directory* é o diretório da biblioteca do cliente da origem de dados.

Para versões de 64 bits, certifique-se de configurar a variável de ambiente DB2LIBPATH para o diretório `lib` de 64 bits.

Exemplo

Para o driver ODBC de 64 bits do Netezza para UNIX:

```
db2set
DB2LIBPATH=/<netezza_odbc_install_path>netezza/lib64
```

3. Para os sistemas operacionais Linux e UNIX, os seguintes itens deverão ser configurados no seu servidor federado:
 - Inclua o diretório de biblioteca do cliente de origem de dados no DB2LIBPATH em `db2dj.ini`:


```
DB2LIBPATH=/<netezza_odbc_install_path>netezza/lib64
```

É necessário reciclar a instância do DB2 para que as mudanças entrem em vigor.
 - Crie um link simbólico para `odbc.ini`:


```
ln -s $ODBCINI $HOME/.odbc.ini
```
4. Registre o wrapper ODBC.
5. Registre as definições do servidor para uma origem de dados ODBC.
6. Crie um mapeamento de usuário para uma origem de dados ODBC.
7. Teste a conexão com o servidor da origem de dados ODBC.
8. Registre apelidos para as tabelas e visualizações da origem de dados ODBC.

Configurando o Acesso às Origens de Dados do DB OLE

Para configurar o servidor federado para acessar origens de dados OLE DB, você deve fornecer ao servidor federado informações sobre os fornecedores OLE DB.

Antes de Iniciar

- A federação deve ser instalada em um servidor que atue como o servidor federado.
- Verifique a configuração do servidor federado.
- Verifique o parâmetro federado para assegurar que a federação está ativada.

Sobre Esta Tarefa

O Microsoft OLE DB é um conjunto de interfaces COM (Component Object Model) que permite que aplicativos acessem informações que normalmente não são armazenados em bancos de dados. Um consumidor OLE DB é qualquer parte do sistema ou código do aplicativo que consome uma interface OLE DB. Um fornecedor OLE DB é qualquer componente de software que expõe uma interface OLE DB.

Você pode configurar um servidor federado para acessar dados que estão armazenados nas origens de dados OLE DB emitindo instruções SQL na linha de comandos do DB2.

Após configurar o acesso à origem de dados OLE DB, utilize a instrução CREATE FUNCTION para registrar uma função de tabela externa OLE DB definida pelo usuário no banco de dados federado.

Procedimento

1. Registre o wrapper.
2. Registre a definição do servidor.
3. Crie os mapeamentos de usuário.

Registrando o Wrapper OLE DB

Você deve registrar um wrapper para acessar origens de dados OLE DB. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Sobre Esta Tarefa

O wrapper OLE DB permite que você acesse fornecedores OLE DB que estão em conformidade com o Microsoft OLE DB versão 2.0 (ou posterior).

O wrapper OLE DB é utilizado apenas para auxiliá-lo no registro de funções de tabela externas OLE DB definidas pelo usuário no banco de dados federado. Diferente de outros wrappers, o wrapper OLE DB não utiliza apelidos para acessar dados que estão armazenados nas origens de dados.

Procedimento

Emita a instrução CRIAR WRAPPER a partir da linha de comandos e especifique o nome padrão para o wrapper OLE DB.

Por Exemplo:

```
CREATE WRAPPER OLEDB
```

Lembre-se: Ao registrar o wrapper utilizando o nome padrão, OLEDB, o servidor federado utiliza automaticamente a biblioteca de wrapper OLE DB apropriada para o sistema operacional no qual seu servidor federado está em execução.

Se o nome padrão do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, você poderá substituir o nome padrão do wrapper por um de sua escolha. Se você não utilizar o nome padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.

Por exemplo, para registrar um wrapper com o nome oledb_wrapper em um servidor federado que utiliza o Windows, emita a seguinte instrução:

```
CREATE WRAPPER oledb_wrapper  
LIBRARY 'db2oledb.dll'
```

O arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado.

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar a definição do servidor.

Arquivos da Biblioteca do Wrapper OLE DB

Os arquivos da biblioteca do wrapper OLE DB são incluídos no servidor federado ao instalar o wrapper.

Ao instalar o wrapper OLE DB, o arquivo de biblioteca é incluído no caminho de diretório padrão.

Se você não utilizar o nome do wrapper padrão ao registrá-lo, você deve incluir o parâmetro LIBRARY na instrução CREATE WRAPPER e especifique o nome do arquivo de biblioteca.

O caminho de diretório padrão e o nome do arquivo de biblioteca do wrapper padrão estão listados na tabela a seguir.

Tabela 31. Locais de Biblioteca e Nomes de Arquivos do Cliente OLE DB

Sistema Operacional	Caminho do Diretório	Nome do Arquivo de Biblioteca
Windows	%DB2PATH%\bin	db2oledb.dll

Registrando as Definições de Servidor para uma Origem de Dados OLE DB

Você deve registrar cada servidor de origem de dados OLE DB que deseja acessar no banco de dados federado.

Sobre Esta Tarefa

Você registra uma definição de servidor para e uma origem de dados OLE DB na linha de comandos do DB2.

Procedimento

Emita a instrução CREATE SERVER.

Por exemplo:

```
CREATE SERVER server_definition_name WRAPPER wrapper_name  
OPTIONS (CONNECTSTRING 'keyword=value;keyword=value');
```

Embora a variável CONNECTSTRING esteja especificada como uma opção na instrução CREATE SERVER, esta opção é requerida para as origens de dados OLE DB.

O que Fazer Depois

Após concluir esta tarefa, você pode criar um mapeamento de usuário.

Instrução CREATE SERVER - Exemplos para o Wrapper OLE DB

Utilize a instrução CREATE SERVER para registrar definições de servidor para o wrapper OLE DB. Este tópico fornece um exemplo completo com os parâmetros requeridos e um exemplo com opções de servidor adicionais.

O exemplo a seguir mostra como registrar uma definição de servidor para um wrapper OLE DB emitindo a instrução CREATE SERVER:

```
CREATE SERVER Nwind WRAPPER OLEDB  
OPTIONS (CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;  
Data Source=c:\msdasdk\bin\oledb\nwind.mdb');
```


Nwind Um nome que você designa à origem de dados OLE DB. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER OLEDB

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

CONNECTSTRING '*Provider=Microsoft.Jet.OLEDB.4.0; Data Source=c:\msdasdk\bin\oledb\nwind.mdb*'

Fornece as propriedades de inicialização que são necessárias para conexão com o fornecedor OLE DB.

O valor para a opção CONNECTSTRING contém uma série de pares de palavra-chave e valor que são separados por pontos-e-vírgulas. O sinal de igual (=) separa cada palavra-chave e seu valor. As palavras-chave são as descrições das propriedades de inicialização OLE DB (conjunto de propriedades DBPROPSET_DBINT) ou palavras-chave específicas do fornecedor.

Para obter uma sintaxe e semântica completas da opção CONNECTSTRING, consulte a publicação *Microsoft OLE DB 2.0 Programmer's Reference and Data Access SDK*, Microsoft Press, 1998.

Opções do Servidor

Ao criar a definição do servidor, você pode especificar opções de servidor adicionais na instrução CREATE SERVER. As opções de servidor podem ser opções de servidor gerais e opções de servidor específicas do OLE DB.

A opção de servidor COLLATING_SEQUENCE especifica se a origem de dados utiliza a mesma seqüência de intercalação que o servidor federado. A configuração padrão para a opção de servidor COLLATING_SEQUENCE é 'N'. Quando a origem de dados OLE DB utiliza a mesma seqüência de intercalação que o servidor federado, a configuração da opção de servidor COLLATING_SEQUENCE é 'Y'.

O seguinte exemplo apresenta uma definição de servidor OLE DB que inclui a opção de servidor COLLATING_SEQUENCE:

```
CREATE SERVER Nwind WRAPPER OLEDB
  OPTIONS (CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;
    Data Source=c:\msdasdk\bin\oledb\nwind.mdb',
  COLLATING_SEQUENCE 'Y')
```

Criando os Mapeamentos de Usuário para uma Origem de Dados OLE DB

Ao tentar acessar um servidor OLE DB, o servidor federado estabelece uma conexão com o servidor OLE DB utilizando um ID de usuário e senha que são válidos para a origem de dados. Você deve definir uma associação (um mapeamento de usuário) entre cada ID e senha de autorização federada e ID de usuário e senha correspondentes da origem de dados.

Procedimento

Emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR local_userid SERVER server_definition_name
  OPTIONS (REMOTE_AUTHID 'remote_userID', REMOTE_PASSWORD 'remote_password');
```

Se o comprimento da senha na origem de dados OLE DB ou da senha no servidor federado é menor do que oito caracteres, as instruções SQL que acessam a origem de dados OLE DB falharão. A seguinte mensagem de erro será exibida:

```
SQL30082N A tentativa de estabelecer uma conexão falhou com a razão de segurança "15"
("PROCESSANDO FALHA"). SQLSTATE=08001
```

Para evitar este problema, altere a senha da origem de dados OLE DB ou altere a senha no servidor federado para oito ou mais caracteres.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper OLE DB

Utilize a instrução CREATE USER MAPPING para mapear um ID de autorização federado para um ID de usuário e senha do OLE DB remoto. Este tópico oferece um exemplo completo com os parâmetros necessários e um exemplo que mostra como utilizar o USER de registro especial do DB2 com a instrução CREATE USER MAPPING.

O exemplo a seguir mostra como mapear um ID de autorização federado para um ID de usuário e senha de uma origem de dados OLE remota:

```
CREATE USER MAPPING FOR laura SERVER Nwind
  OPTIONS (REMOTE_AUTHID 'lulu', REMOTE_PASSWORD 'raiders')
```

laura Especifica o ID de autorização local que você está mapeando para um ID de usuário e senha remotos, que estão definidos na origem de dados OLE DB.

SERVER *Nwind*

Especifica o nome da definição do servidor que você registrou na instrução CREATE SERVER para o servidor OLE DB.

REMOTE_AUTHID '*lulu*'

Especifica o ID do usuário no servidor de banco de dados OLE DB para o qual você está mapeando *lulu*. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_ID para 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD '*raiders*'

Especifica a senha que está associada com '*lulu*'. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

USER de Registro Especial do DB2

Você pode utilizar o USER de registro especial do DB2 para mapear o ID de autorização da pessoa que está emitindo a instrução CREATE USER MAPPING para o ID de autorização da origem de dados especificado na opção de usuário REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER SERVER Nwind
  OPTIONS (REMOTE_AUTHID 'lulu', REMOTE_PASSWORD 'raiders');
```

Configurando o Acesso às Origens de Dados do Oracle

Para configurar um sistema federado para acessar origens de dados Oracle, você deve fornecer ao sistema federado informações sobre as origens de dados e os objetos que deseja acessar.

Antes de Iniciar

- O software cliente do Oracle deve ser instalado e configurado no servidor que age como servidor federado.
- A federação deve ser instalada em um servidor que atue como o servidor federado.
- Verifique a configuração do servidor federado.
- Verifique o parâmetro federado para assegurar que a federação está ativada.

Procedimento

1. Configure as variáveis de ambiente do Oracle.
2. Configuração e Teste do Arquivo de Configuração do Cliente Oracle
3. Registro do Wrapper do Oracle.
4. Registro das Definições do Servidor para uma Origem de Dados do Oracle.
5. Crie os mapeamentos de usuário.
6. Teste a conexão com o servidor Oracle.
7. Registre os apelidos.

Definindo as Variáveis de Ambiente do Oracle

As variáveis de ambiente do Oracle devem ser configuradas no arquivo `db2dj.ini` no servidor federado.

Sobre Esta Tarefa

Restrições

Revise as restrições ao arquivo `db2dj.ini`.

O arquivo `db2dj.ini` contém informações de configuração sobre o software cliente Oracle que está instalado no servidor federado.

Existem variáveis de ambiente requeridas e opcionais para origens de dados Oracle.

Se você instalou o software cliente do Oracle antes de instalar o wrapper do Oracle, as variáveis de ambiente necessárias do Oracle estarão configuradas no arquivo `db2dj.ini`.

Se o software cliente do Oracle não foi instalado antes da instalação do wrapper do Oracle ou caso você deseja configurar qualquer uma das variáveis de ambiente, é preciso configurar as variáveis de ambiente usando as etapas nessa tarefa.

Procedimento

1. Utilize um dos seguintes métodos:

Método	Descrição
Configurar automaticamente as variáveis de ambiente.	Execute o assistente de instalação do IBM InfoSphere Federation Server. Siga as instruções no assistente. Importante: Configure as variáveis de ambiente necessárias executando o assistente de instalação. As variáveis de ambiente opcionais devem ser definidas manualmente.

Método	Descrição
Configure as variáveis de ambiente com a ferramenta de configuração do wrapper.	Execute a ferramenta de configuração do wrapper para configurar as variáveis de ambiente necessárias e as opcionais. Siga as instruções em Executando a ferramenta de configuração do wrapper.
Configurar manualmente as variáveis de ambiente.	<p>Edite o arquivo <code>db2dj.ini</code>:</p> <p>O arquivo <code>db2dj.ini</code> está localizado no diretório especificado pela variável de registro <code>DB2_DJ_INI</code> do DB2. Quando a variável <code>DB2_DJ_INI</code> não está configurada, o arquivo <code>db2dj.ini</code> corresponde a um dos seguintes caminhos padrão, dependendo do sistema operacional:</p> <ul style="list-style-type: none"> Em Linux e UNIX: <code>instancehome/sqllib/cfg/db2dj.ini</code>. <i>instancehome</i> O diretório inicial do proprietário da instância. Em Windows: <code>%DB2PATH%\cfg\db2dj.ini</code> <code>%DB2PATH%</code> O diretório em que o sistema de banco de dados do DB2 está instalado, por exemplo, <code>C:\Program Files\IBM\sqllib</code>. <p>Se o arquivo não existir, você poderá criar um arquivo com o nome <code>db2dj.ini</code> utilizando qualquer editor de texto. No arquivo <code>db2dj.ini</code>, você deve especificar o caminho completo no valor das variáveis de ambiente; caso contrário, encontrará erros.</p>

- Em servidores federados que executam em UNIX, inclua a variável de ambiente Oracle no arquivo `.profile` da instância do DB2. Por exemplo:

```
export ORACLE_HOME=oracle_home_directory
export PATH=$ORACLE_HOME/bin:$PATH
```

Em que *oracle_home_directory* é o diretório no qual o software do cliente Oracle está instalado.

- Em servidores federados que executam em Linux ou UNIX, execute o arquivo `.profile` da instância do DB2 inserindo:

```
.$HOME/.profile
```

- Para assegurar que as variáveis de ambiente estejam configuradas no servidor federado, recicle a instância do DB2 com esses comandos:

```
db2stop
db2start
```

O que Fazer Depois

Depois de concluir essa tarefa, é possível configurar e testar o cliente do Oracle.

Variáveis de Ambiente do Oracle

Existem variáveis de ambiente requeridas e opcionais para origens de dados Oracle. Essas variáveis são definidas no arquivo `db2dj.ini`.

As variáveis de ambiente a seguir são válidas para o Oracle:

- ORACLE_HOME
- ORACLE_BASE (opcional)
- ORA_NLS (opcional)
- NLS_LANG (opcional)

- TNS_ADMIN (opcional)

Descrições da Variável

ORACLE_HOME

Configure a variável de ambiente ORACLE_HOME para o caminho do diretório em que o software do cliente Oracle está instalado. Especifique o caminho completo para a variável de ambiente:

ORACLE_HOME=*oracle_home_directory*. Por exemplo, se o diretório inicial do Oracle for `\usr\oracle\8.1.7`, a entrada no arquivo `db2dj.ini` será `ORACLE_HOME=\usr\oracle\8.1.7`.

Se o usuário individual da instância federada não configurar a variável de ambiente ORACLE_HOME localmente, a instância federada não utiliza essa configuração. A instância federada usa apenas o valor de ORACLE_HOME configurado no arquivo `db2dj.ini`.

ORACLE_BASE

ORACLE_BASE representa a raiz da árvore do diretório do cliente Oracle. Se a variável de ambiente ORACLE_BASE for configurada durante a instalação do software do cliente Oracle, configure a variável de ambiente ORACLE_BASE no servidor federado.

Por exemplo:

`ORACLE_BASE=oracle_root_directory`

ORA_NLS

Se diversas versões do Oracle estiverem em execução em seu sistema, você deve garantir que:

- A variável de ambiente ORA_NLS apropriada esteja configurada
- Os arquivos de dados de NLS correspondentes para as versões que você está usando estejam disponíveis

Os dados específicos do local são armazenados em um diretório especificado pela variável de ambiente ORA_NLS. Cada versão do Oracle possui um diretório de dados ORA_NLS diferente.

Tabela 32. Variável de Ambiente ORA_NLS por Versão

Versão do Oracle	Variável de Ambiente
8.x, 9.x	ORA_NLS33
10.x	ORA_NLS10

Por exemplo, em servidores federados que executam em UNIX e acessam as origens de dados Oracle 8.1, a configuração da variável de ambiente ORA_NLS33 é:

`ORA_NLS33=oracle_home_directory/ocommon/nls/admin/<data>`

NLS_LANG

A variável de ambiente NLS_LANG representa a variável de ambiente da página de códigos. Consulte a documentação do Oracle NLS para obter informações sobre a configuração desta variável.

TNS_ADMIN

No servidor federado com Windows

O cliente do Oracle procura pelo arquivo `tnsnames.ora` no diretório `%ORACLE_HOME%\NETWORK\ADMIN`, em que `%ORACLE_HOME%` é definido no arquivo `db2dj.ini`. Se o arquivo `tnsnames.ora` não estiver no

diretório %ORACLE_HOME%\NETWORK\ADMIN, você deve configurar a variável de ambiente TNS_ADMIN no arquivo the db2dj.ini no servidor federado. Você configura a variável de ambiente no arquivo db2dj.ini para o caminho e que o arquivo tnsnames.ora está localizado.

Em servidores federados que executam em AIX ou Linux

O cliente do Oracle procura pelo arquivo tnsnames.ora no diretório /etc. Se o arquivo tnsnames.ora não estiver no diretório /etc, o cliente do Oracle procura pelo arquivo tnsnames.ora no diretório \$ORACLE_HOME/network/admin, em que \$ORACLE_HOME é definido no arquivo db2dj.ini. Se o arquivo tnsnames.ora não estiver no diretório \$ORACLE_HOME/network/admin, você deve configurar a variável de ambiente TNS_ADMIN no servidor federado. Você configura a variável de ambiente no arquivo db2dj.ini para o caminho e que o arquivo tnsnames.ora está localizado.

Por exemplo, se o arquivo tnsnames.ora está no diretório /home/oracle, é possível configurar a variável de ambiente para:

```
TNS_ADMIN=/home/oracle
```

Em servidores federados que executam em Solaris

O cliente do Oracle procura pelo arquivo tnsnames.ora no diretório /var/opt/oracle. Se o arquivo tnsnames.ora não estiver no diretório /var/opt/oracle, o cliente do Oracle procura pelo arquivo tnsnames.ora no diretório \$ORACLE_HOME/network/admin, em que \$ORACLE_HOME é definido no arquivo db2dj.ini. Se o arquivo tnsnames.ora não estiver no diretório \$ORACLE_HOME/network/admin, você deve configurar a variável de ambiente TNS_ADMIN. Você configura a variável no arquivo db2dj.ini para o caminho e que o arquivo tnsnames.ora está localizado.

Por exemplo, se o arquivo tnsnames.ora está no diretório /home/oracle, é possível configurar a variável de ambiente para:

```
TNS_ADMIN=/home/oracle
```

Conversão da Página de Códigos Oracle:

Cada vez que o wrapper do Oracle se conecta a uma origem de dados Oracle, o wrapper determina qual valor de página de códigos utilizar para essa conexão. Você pode fazer com que o wrapper do Oracle defina o valor da página de códigos ou designar uma página de códigos definindo a variável de ambiente NLS_LANG.

As variáveis de ambiente que especificam a conversão da página de códigos do Oracle estão definidas no arquivo db2dj.ini no servidor federado.

Se a variável NLS_LANG estiver definida no arquivo db2dj.ini no servidor federado, o wrapper utiliza o valor da página de códigos no arquivo db2dj.ini.

Se a variável NLS_LANG não estiver definida no arquivo db2dj.ini no servidor federado, o wrapper determina o território e a página de códigos do banco de dados federado. O wrapper configura a variável de ambiente NLS_LANG para o código do idioma que melhor corresponde ao Oracle. Se não houver nenhum código do idioma de correspondência aproximada, a variável de ambiente NLS_LANG é configurada para American_America.US7ASCII.

Consulte a documentação que acompanha seu software Oracle para uma lista de códigos de idiomas válidos.

Exemplo de Página de Códigos Chineses GB 18030:

Caso você acesse uma origem de dados Oracle que contenha dados codificados com a página de códigos Chinesa GB 18030 e seu banco de dados federado usa a página de códigos UTF-8; o wrapper do Oracle configura a variável de ambiente do Oracle NLS_LANG para:

```
NLS_LANG=Simplified Chinese_China.UTF8
```

Essa configuração está correta se você estiver usando um cliente Oracle 8, as se você estiver usando o cliente do Oracle Versão 9i (ou posterior), é preciso configurar a variável de ambiente NLS_LANG para substituir a configuração padrão do wrapper do Oracle. Configure a variável de ambiente NLS_LANG para Simplified Chinese_China.AL32UTF8, de forma que o cliente Oracle 9i converte os dados GB 18030 em Unicode corretamente.

Por exemplo:

```
NLS_LANG=Simplified Chinese_China.AL32UTF8
```

Configurar e Testar o Arquivo de Configuração do Cliente Oracle

O arquivo de configuração do cliente Oracle é utilizado para conectar-se a bancos de dados Oracle utilizando as bibliotecas do cliente instaladas no sistema federado.

Sobre Esta Tarefa

O arquivo de configuração do cliente especifica o local de cada servidor de banco de dados Oracle tipo de conexão (protocolo) para o servidor de banco de dados.

O nome padrão para o arquivo de configuração do cliente Oracle é `tnsnames.ora`.

O local padrão do arquivo de configuração do cliente depende do sistema operacional utilizado pelo sistema federado:

- Em sistemas Linux e UNIX, o local padrão do arquivo é `$ORACLE_HOME/network/admin`.
- Em sistemas Windows, o local padrão do arquivo é `%ORACLE_HOME%\NETWORK\ADMIN`.

Procedimento

1. Crie um `tnsnames.ora` usando o Utilitário de configuração Oracle NET8/NET que acompanha o software do cliente Oracle.
Dentro do arquivo `tnsnames.ora`, o SID (ou `SERVICE_NAME`) corresponde ao nome da instância do Oracle e o `HOST` corresponde ao nome do host em que o servidor Oracle está localizado.
2. Se desejar colocar o arquivo `tnsnames.ora` em um caminho que não seja o caminho da procura padrão, defina a variável de ambiente `TNS_ADMIN` para especificar o local do arquivo:
 - a. Edite o arquivo `db2dj.ini` no diretório `sqllib/cfg` e configure a variável de ambiente `TNS_ADMIN`. Por exemplo:

```
TNS_ADMIN=x:/path/
```
 - b. Emita os seguintes comandos para reciclar a instância do DB2 e garantir que a variável de ambiente esteja configurada no programa:

```
db2stop  
db2start
```

3. Teste a conexão para garantir que o software do cliente pode ser conectado ao servidor Oracle. Utilize o utilitário sqlplus do Oracle para testar a conexão.

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar o wrapper do Oracle.

Registrando o Wrapper Oracle

Você deve registrar um wrapper para acessar origens de dados Oracle. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Procedimento

Use um dos métodos a seguir para registrar um wrapper do Oracle:

Método	Descrição
Use o Explorer de Administração no Data Studio.	Na pasta Objetos de Banco de Dados Federado , clique com o botão direito do mouse na pasta Wrappers e selecione Criar Wrapper .
Emita a instrução CREATE WRAPPER e especifique o nome padrão para o wrapper do Oracle.	<p>Por exemplo:</p> <pre>CREATE WRAPPER NET8 ;</pre> <p>Lembre-se: Ao registrar um wrapper utilizando o nome padrão, NET8, o servidor federado utiliza automaticamente a biblioteca de wrapper apropriada do Oracle para o sistema operacional no qual o servidor federado está em execução.</p> <p>Se não utilizar o nome padrão do wrapper ao registrá-lo, você deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER e especificar o nome do arquivo da biblioteca padrão do wrapper.</p> <p>Por exemplo, para registrar um wrapper com o nome de oracle_wrapper em um servidor federado que utiliza o AIX, emita a seguinte instrução:</p> <pre>CREATE WRAPPER oracle_wrapper LIBRARY 'libdb2net8.a'</pre> <p>O arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado.</p>

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar as definições do servidor.

Arquivos de Biblioteca do Wrapper Oracle

Os arquivos de biblioteca do wrapper do Oracle são incluídos no servidor federado ao instalar o wrapper.

Ao instalar o wrapper do Oracle, três arquivos de biblioteca são incluídos no caminho padrão do diretório. Por exemplo, se um servidor federado estiver em sendo executado em AIX, os arquivos da biblioteca de wrapper incluídos no caminho de diretório a seguir são libdb2net8.a, libdb2net8F.a e libdb2net8U.a, O arquivo de biblioteca padrão é libdb2net8.a. Os outros arquivos de biblioteca de wrapper são usados internamente pelo wrapper do Oracle.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 33. Locais de Biblioteca do Wrapper e Nomes de Arquivos do Oracle

Sistema Operacional	Caminho do Diretório	Nomes do Arquivo de Biblioteca
AIX	/usr/opt/install_path/lib32/	libdb2net8.a
	/usr/opt/install_path/lib64/	
Linux	/opt/IBM/db2/install_path/lib32	libdb2net8.so
	/opt/IBM/db2/install_path/lib64	
Solaris	/opt/IBM/db2/install_path/lib32	libdb2net8.so
	/opt/IBM/db2/install_path/lib64	
Windows	%DB2PATH%\bin	db2net8.dll

install_path é o caminho do diretório em que servidor federado está instalado no UNIX ou Linux.

Registrando Definições de Servidor para uma Origem de Dados Oracle

Você deverá registrar cada servidor Oracle que deseja acessar no banco de dados federado.

Procedimento

1. Localize o nome do nó no arquivo tnsnames.ora do Oracle. Exemplo de arquivo tnsnames.ora:

```

paris_node =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = somehost)(PORT = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = ora9i.see1)))

```

Neste exemplo, o valor do nó a ser utilizado na instrução CREATE SERVER é *paris_node*.

Embora *node_name* esteja especificado como uma opção na instrução CREATE SERVER SQL, é exigido para origens de dados Oracle.

2. para registrar as definições do servidor para um origem de dados Oracle, use um dos seguintes métodos:

Método	Descrição
Use o Explorer de Administração no Data Studio.	Na pasta Objetos de Banco de Dados Federado , clique com o botão direito do mouse na pasta Servidores Remotos e selecione Criar Servidor Remoto .

Método	Descrição
Emita a instrução CREATE SERVER.	<p>Por exemplo:</p> <pre>CREATE SERVER server_name TYPE oracle VERSION 8.1.7 WRAPPER net8 OPTIONS (NODE 'node_name')</pre> <p>Após a criação da definição do servidor, utilize a instrução ALTER SERVER para incluir ou descartar opções do servidor.</p>

O que Fazer Depois

A tarefa seguinte nessa seqüência de tarefas é criar os mapeamentos de usuários para uma origem de dados Oracle.

Instrução CREATE SERVER - Exemplos para o Wrapper do Oracle

Utilize a instrução CREATE SERVER para registrar definições de servidor para o wrapper do Oracle. Este tópico fornece um exemplo completo com os parâmetros requeridos e um exemplo com opções de servidor adicionais.

O exemplo a seguir mostra como registrar uma definição de servidor para um wrapper do Oracle emitindo a instrução CREATE SERVER:

```
CREATE SERVER oraserver TYPE oracle VERSION 8.1.7 WRAPPER wrapper_name
OPTIONS (NODE 'paris_node') ;
```

oraserver

Um nome que você atribui ao servidor de banco de dados Oracle. Nomes de definição de servidor duplicados não são permitidos.

TYPE *oracle*

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso. Para o wrapper do Oracle, o tipo de servidor deve ser *oracle*.

VERSION *8.1.7*

A versão do servidor de banco de dados Oracle que você deseja acessar.

WRAPPER *wrapper_name*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

NODE *'paris_node'*

O nome do nó no qual o servidor de bancos de dados Oracle reside. Obtenha o nome do nó a partir do arquivo *tnsnames.ora*. Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é necessário para origens de dados Oracle.

Opções do Servidor

Ao criar a definição do servidor, você pode especificar opções de servidor adicionais na instrução CREATE SERVER. Essas opções do servidor podem ser gerais ou específicas do Oracle.

O servidor federado assume que todas as colunas VARCHAR do Oracle contém espaços em branco finais. Caso esteja certo de que todas as colunas de VARCHAR no banco de dados Oracle não contiverem espaços em branco finais, é possível

configurar uma opção do servidor para especificar que a origem de dados usa semântica de comparação VARCHAR padded que não está em branco.

O exemplo a seguir mostra uma definição de servidor oracle com a opção de servidor VARCHAR_NO_TRAILING_BLANKS:

```
CREATE SERVER oraserver TYPE oracle VERSION 8.1.7 WRAPPER wrapper_name
  OPTIONS (NODE 'paris_node', VARCHAR_NO_TRAILING_BLANKS 'Y') ;
```

Use a opção do servidor VARCHAR_NO_TRAILING_BLANKS quando nenhuma das colinas contiver espaços em branco finais. Caso apenas algumas das colunas VARCHAR contiverem espaços em branco finais, é possível configurar uma opções sobre as referidas colunas com a instrução ALTER NICKNAME.

Criando Mapeamentos de Usuários para uma Origem de Dados Oracle

para acessar um servidor Oracle, o servidor federado estabelece uma conexão com o servidor Oracle usando um ID de usuário e senha válidos para aquela origem de dados.

Sobre Esta Tarefa

Restrições

O ID do usuário na origem de dados Oracle deve ser criados usando o comando do Oracle **create user** com a cláusula *identified by*, ao invés da clausula *identificada externamente*.

Procedimento

Use um dos métodos a seguir para criar os mapeamentos de usuário para uma origem de dados Oracle:

Método	Descrição
Use o Explorador de Administração no Data Studio.	Na pasta Objetos de Banco de Dados Federado , clique com o botão direito do mouse na pasta Mapeamentos de Usuário e selecione Criar Mapeamentos de Usuário .
Emita a instrução CREATE USER MAPPING.	Por exemplo: <pre>CREATE USER MAPPING FOR local_userID SERVER server_definition_name OPTIONS (REMOTE_AUTHID 'remote_userID', REMOTE_PASSWORD 'remote_password') ;</pre>

O que Fazer Depois

Depois de concluir essa tarefa, teste a conexão com o servidor Oracle.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper do Oracle

Utilize a instrução CREATE USER MAPPING para mapear um ID de autorização federado do para um ID de usuário e senha do servidor remoto. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o USER de registro especial do DB2 com a instrução CREATE USER MAPPING.

O exemplo a seguir mostra como mapear um ID de autorização federado para um ID de usuário e senha do Oracle:

```
CREATE USER MAPPING FOR robert SERVER oraserver
  OPTIONS (REMOTE_AUTHID 'rob', REMOTE_PASSWORD 'then4now') ;
```

robert Especifica o ID de autorização que você está mapeando para um ID de usuário e senha remotos, definidos no servidor Oracle.

SERVER *oraserver*

Especifica o nome da definição do servidor registrado na instrução CREATE SERVER para o servidor Oracle.

REMOTE_AUTHID '*rob*'

Especifica o ID de usuário remoto para o qual *robert* está sendo mapeado. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_ID para 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD '*then4now*'

Especifica a senha remota associada a '*al*'. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

USER de Registro Especial do DB2

Você pode utilizar o USER de registro especial do DB2 para mapear o ID de autorização da pessoa que está emitindo a instrução CREATE USER MAPPING para o ID de autorização da origem de dados especificado na opção de usuário REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER SERVER oraserver
  OPTIONS (REMOTE_AUTHID 'rob', REMOTE_PASSWORD 'then4now') ;
```

Mapeamento de Usuário Público

No InfoSphere Federation Server Versão 9.7 Fix Pack 2 e posteriores, é possível criar um mapeamento de usuário público para permitir que todos os usuários do banco de dados local acessem uma origem de dados por meio de um único ID de usuário e uma única senha remotos.

Neste exemplo, um wrapper net8 e um servidor de origem de dados do Oracle oraserver são criados. Em seguida, a instrução CREATE USER MAPPING é especificada com PUBLIC para criar um mapeamento de usuário público para o servidor oraserver e mapear todos os usuários do banco de dados local para o ID de autorização que é especificado nas OPTIONS da instrução CREATE USER MAPPING.

Exemplo

```
CREATE WRAPPER net8;

CREATE SERVER oraserver
  TYPE oracle VERSION 8.1.7 WRAPPER net8
  OPTIONS (NODE 'paris_node') ;

CREATE USER MAPPING FOR PUBLIC SERVER oraserver
  OPTIONS (REMOTE_AUTHID 'APP_USER', REMOTE_PASSWORD 'secret');
```

Testando a Conexão com o Servidor Oracle

Teste a conexão com o servidor Oracle para determinar se o servidor federado está configurado adequadamente para acessar as origens de dados Oracle.

Sobre Esta Tarefa

Você pode testar a conexão com o servidor Oracle utilizando a definição do servidor e os mapeamentos de usuário que definiu.

Procedimento

Abra uma sessão de passagem e emita uma instrução SELECT nas tabelas de sistema Oracle. Se a instrução SELECT retornar uma contagem, a definição do servidor e o mapeamento de usuário estão configurados apropriadamente.

Por exemplo:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM sys.all_tables
SET PASSTHRU RESET
```

Se a instrução SELECT retornar um erro, você deverá resolver os problemas dos erros de conexão.

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar os apelidos das tabelas e visualizações do Oracle.

Resolução de Problemas de Conectividade com Origens de Dados Oracle

O problema mais comum que pode ser encontrado ao configurar o servidor federado para acessar as origens de dados Oracle é a conectividade.

Sintoma

Caso não seja possível se conectar a uma origem de dados Oracle a partir de um servidor federado, você pode precisar atualizar o arquivo hosts de TCP/IP.

Causa

Esse problema pode ser causado por um arquivo hosts de TCP/IP desatualizado.

Solução do Problema

Para cada host na seção DESCRIPTION do arquivo tnsnames.ora, pode ser preciso atualizar o arquivo hosts de TCP/IP. A atualização desse arquivo vai depender de como o TCP/IP está configurado em sua rede. Parte da rede deve ser convertida no nome de host remoto especificado na seção DESCRIPTION no arquivo tnsnames.ora para um endereço.

Caso sua rede tenha um servidor de nomes que reconheça o nome do host, não é preciso atualizar o arquivo hosts de TCP/IP. Caso sua rede não tenha um servidor de nomes que reconheça o nome do host, é preciso incluir uma entrada no arquivo hosts de TCP/IP para o host remoto.

O local do arquivo hosts de TCP/IP depende do sistema operacional em execução no servidor federado:

Em servidores federados que executam em Linux ou UNIX

O arquivo hosts encontra-se no diretório /etc/hosts.

No servidor federado com Windows

O arquivo hosts encontra-se no diretório x:\winnt\system32\drivers\etc\hosts.

Resolução de Problemas de Erros de Conexão da Origem de Dados

Uma conexão de teste com o servidor de origem de dados pode retornar um erro por diversas razões. Há ações que você pode executar para determinar a causa de um erro.

Sintoma

É retornado um erro quando você tentar se conectar à origem de dados.

Causa

Há diversas causas possíveis para um problema de conexão.

Solução do Problema

Para resolver problemas de erros de conexão com a origem de dados, verifique os seguintes itens:

- Verifique se a origem de dados está disponível.
- Se aplicável, certifique-se de que o servidor de origem de dados esteja configurado para conexões de chegada.
- Certifique-se de suas configurações de mapeamento de usuário para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com a origem de dados. Altere o mapeamento de usuário ou crie outro, conforme necessário.
- Se aplicável, certifique-se de que o software do cliente da origem de dados no servidor federado esteja instalado e configurado corretamente para conexão com a origem de dados.
- Para origens de dados ODBC, certifique-se de que o driver ODBC no servidor federado esteja instalado e configurado corretamente para conexão com o servidor de origem de dados ODBC. Em servidores federados que executam Windows, utilize a ferramenta Administrador de Origem de Dados ODBC para verificar o driver. Em servidores de dados que executam UNIX, consulte a documentação do fornecedor do cliente ODBC.
- Verifique se as configurações para as variáveis configuradas no servidor federado estão corretas para a origem de dados. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis no arquivo db2dj.ini e as variáveis do DB2 Profile Registry (db2set).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e a crie novamente.

Registrando Pseudônimos para Tabelas e Visualizações do Oracle

Para cada definição de servidor Oracle registrada, você deverá registrar um pseudônimo para cada tabela ou visualização que deseja acessar. Utilize esses pseudônimos ao invés dos nomes dos objetos de origem de dados quando consultar os servidores Oracle.

Antes de Iniciar

Atualize as estatísticas na origem de dados Oracle antes de registrar um pseudônimo. O banco de dados federado confia nas estatísticas do catálogo da origem de dados para otimizar o processamento da consulta. Utilize o comando da origem de dados equivalente ao comando do DB2 **RUNSTATS** para atualizar as estatísticas da origem de dados.

Sobre Esta Tarefa

para objetos de dados que usam Oracle Label Security, os dados do apelido não podem ser armazenados em caixa. É possível ligar e desligar o armazenamento em cache usando a instrução ALTER NICKNAME.

Restrição: Criar um apelido em um sinônimo de um sinônimo do Oracle não é suportado e irá lançar a mensagem de erro SQL0204.

Procedimento

Para registrar um apelido para uma tabela ou visualização do Oracle, use um dos métodos a seguir:

Método	Descrição
Use o Explorer de Administração no Data Studio.	Na pasta Objetos de Banco de Dados Federado , clique com o botão direito do mouse na pasta Apelidos e selecione Criar Apelido .
Emita a instrução CREATE NICKNAME.	Por exemplo: CREATE NICKNAME <i>nickname</i> FOR <i>server_definition_name</i> ." <i>remote_schema</i> ". <i>"remote.table"</i> ;

Ao criar o apelido, o servidor federado consulta o catálogo de origem de dados utilizando o apelido. Esta consulta testa a conexão com a tabela ou visualização da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

Repita esta etapa para cada tabela ou visualização do Oracle para as quais deseja criar um apelido.

Instrução CREATE NICKNAME - Exemplos para o Wrapper do Oracle

Utilize a instrução CREATE NICKNAME para registrar um apelido para uma tabela ou visualização do Oracle que deseja acessar. Este tópico fornece um exemplo completo com os parâmetros necessários.

O exemplo a seguir mostra como registrar um apelido para uma tabela ou visualização Oracle usando a instrução CREATE NICKNAME.

```
CREATE NICKNAME PARISINV FOR oraserver."vinnie"."inventory" ;
```

PARISINV

Um apelido exclusivo que é utilizado para identificar a tabela ou visualização do Oracle. O apelido consiste em um esquema e o apelido. Se você omitir o esquema ao registrar o apelido, o esquema do apelido será o ID de autorização do usuário que registrar o apelido.

```
oraserver."vinnie"."inventory"
```

Um identificador de três partes para o objeto remoto:

- *oraserver* é o nome de definição do servidor atribuído ao servidor de banco de dados do Oracle na instrução CREATE SERVER.

- *vinnie* é o ID de usuário do proprietário ao qual a tabela ou visualização pertence.
- *inventory* é o nome da tabela ou visualização remota que você deseja acessar.

O servidor federado altera os nomes dos esquemas e tabelas do Oracle para maiúsculas, a menos que você coloque os nomes entre aspas.

Configurando o Acesso aos Scripts como Origens de Dados

Para configurar um sistema federado para acessar scripts como origens de dados, registre as funções customizadas, um wrapper, uma definição de servidor e apelidos para os scripts.

Antes de Iniciar

- A federação deve ser instalada em um servidor que atue como o servidor federado.
- Um banco de dados federado deve existir no servidor federado.

Sobre Esta Tarefa

É possível configurar um servidor federado para acessar dados por meio de scripts emitindo instruções SQL na linha de comandos do DB2.

Procedimento

1. Identifique ou grave um script.
2. Registre a função customizada.
3. Configure o daemon de scripts.
4. Inicie o daemon de script.
5. Registre o wrapper de script.
6. Registre a definição do servidor para uma origem de dados que é acessada por um script.
7. Registre apelidos para origens de dados.

Visão Geral do Wrapper de Script

É possível utilizar scripts, como scripts Perl, para acessar informações em banco de dados ou para gerar dados. É possível integrar tais dados com dados de outras origens de dados federadas utilizando o wrapper de script.

Você pode ter scripts existentes que retornam dados a partir de origens de dados como bancos de dados de ciências biológicas, ou os dados gerados neles próprios. O wrapper de script permite a utilização de scripts como se eles fossem origens de dados federados. O wrapper de script possibilita acesso aos dados que podem então ser integrados pela utilização de um sistema federado. Os scripts devem retornar resultados em XML.

Os apelidos do wrapper de script podem incluir colunas de entrada e saída. Esses apelidos utilizam modelos de função em predicados para passar valores de entrada para o script. Os dados de saída do script são representados como XML de uma forma hierárquica, os quais podem então ser mapeados para apelidos utilizando-se chaves primárias ou estrangeiras.

O wrapper de script é um wrapper de leitura. O wrapper de script não pode gravar dados em uma origem de dados.

No diagrama a seguir, os fluxos de dados de um script através do wrapper de script e do daemon de script para o banco de dados federado, onde os dados podem ser integrados com dados de outras origens e visualizados com o cliente federado. Opcionalmente, o sistema federado pode acessar o script através de um servidor de proxy e um firewall.

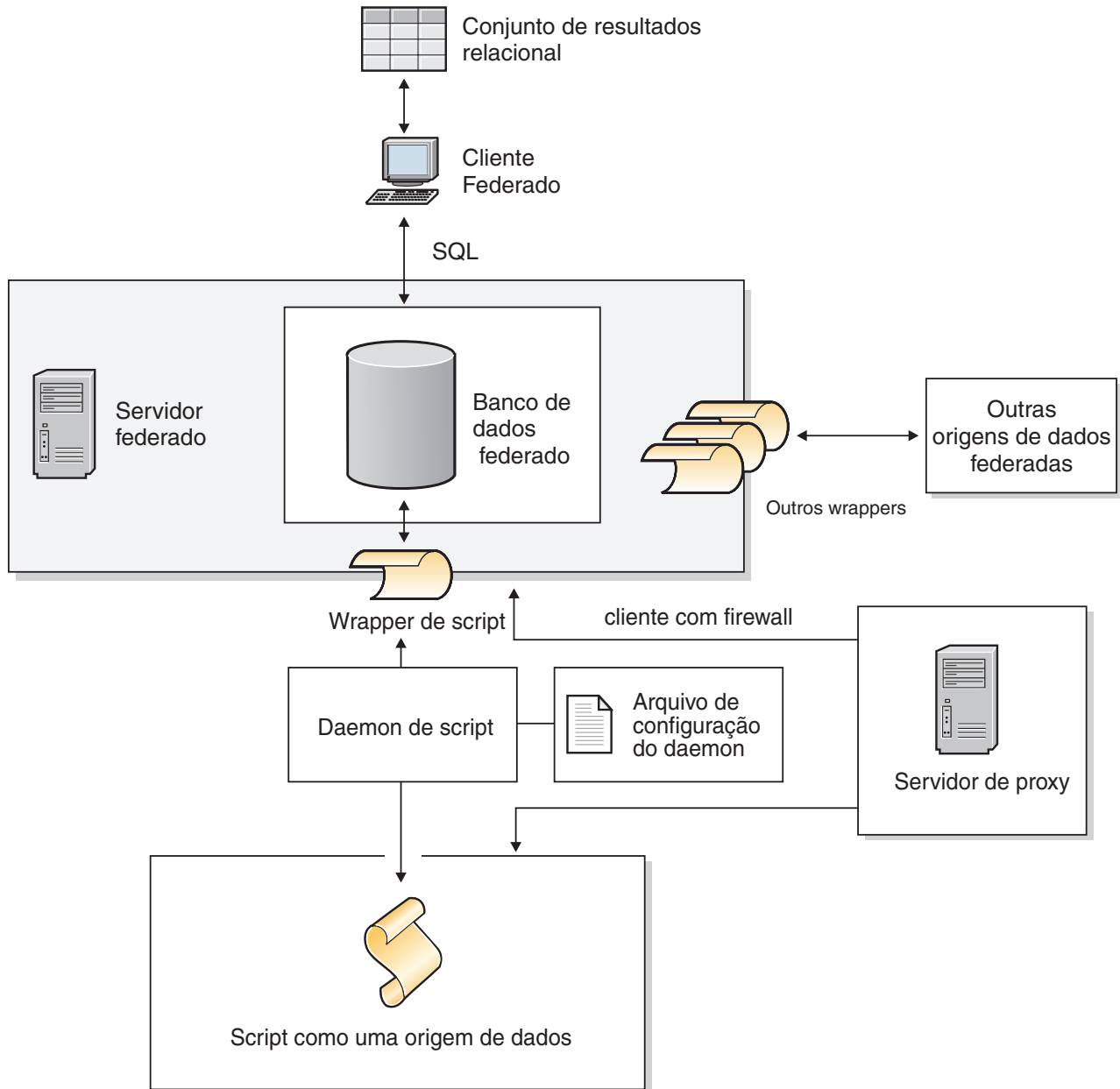


Figura 10. O Wrapper de Script em um Sistema Federado

Um script é chamado a partir do diretório que contém o daemon de scripts. Se o script recuperar dados de seus próprios arquivos de dados e não puder localizar o arquivo de dados, o script pode incluir caminhos relativos. Utilize caminhos absolutos nos scripts.

Incluindo Scripts como Origens de Dados em um Sistema Federado

Para configurar um servidor federado para acessar scripts como origens de dados, configure o daemon de scripts, um wrapper, uma definição de servidor e apelidos para os scripts.

Registrando a função customizada para o script

Registre a função customizada do script WSSCRIPT.ARGS antes de registrar o wrapper de script.

Sobre Esta Tarefa

Registre a função customizada em cada instância do banco de dados federado onde o wrapper de script está instalado.

A função customizada para o wrapper de script deve ser registrada com o nome de esquema WSSCRIPT.

Inclua palavras-chave específicas quando registrar a função customizada para o wrapper de script. Inclua as palavras-chave AS TEMPLATE, DETERMINISTIC e NO EXTERNAL ACTION na instrução CREATE FUNCTION.

O ambiente federado utiliza mecanismos de consulta. Para o wrapper de script, esses mecanismos de consulta são mecanismos de consulta de bancos de dados federados e o mecanismo de consulta do wrapper de script. É possível especificar que predicados foram transferidos para o mecanismo do wrapper de script através das funções customizadas do wrapper de script na cláusula WHERE da sua instrução SELECT.

O arquivo create_function_mappings.ddl no diretório sql/lib/samples/lifesci/script no servidor federado especifica os tipos de dados para a função customizada.

Procedimento

Execute o arquivo create_function_mappings.ddl em cada instância do banco de dados federado onde o wrapper de script está instalado.

Resultados

O exemplo a seguir mostra a sintaxe para a função WSSCRIPT.ARGS:

```
CREATE FUNCTION WSSCRIPT.ARGS (input_column_data_type(), input_column_data_type())
    RETURNS INTEGER AS TEMPLATE
    DETERMINISTIC NO EXTERNAL ACTION;
```

Tipos de Dados para a Função Customizada para o Wrapper de Script:

Especifique o tipo de dados da coluna de entrada duas vezes em cada função customizada.

Registre uma função customizada WSSCRIPT separada para cada tipo de dados válido para o argumento column_name. A função customizada WSSCRIPT possui os seguintes tipos de dados:

- VARCHAR
- INTEGER

- CLOB
- DOUBLE
- DATE

Os parâmetros para a função customizada devem ser do mesmo tipo de dados, que é o tipo de dados da coluna de entrada correspondente. Quando utilizado em um predicado de consulta, o primeiro parâmetro é o nome da coluna de entrada SWITCH. A segunda coluna é o valor a ser passado para o script para este SWITCH.

Configurando o Daemon de Script

O wrapper de script requer um daemon de script que atenda os pedidos de tarefas no wrapper de script. O daemon de scripts deve ser configurado antes do registro do wrapper de script.

Antes de Iniciar

O daemon de script tem os seguintes pré-requisitos:

- Ter acesso de gravação em um diretório no qual o daemon possa gravar arquivos temporários.
- Executar em um servidor que você possa acessar através de TCP/IP a partir de seu sistema federado. Este servidor pode ser o mesmo servidor que opera como o servidor federado ou um servidor de script separado.
- Exige um arquivo de configuração que deve estar no mesmo servidor que o daemon de script.
-
- Executar separadamente a partir do wrapper de script e do banco de dados federado.

Procedimento

1. Certifique-se de que os arquivos executáveis para o daemon de script estejam no servidor correto. Pode ser necessário copiar os arquivos executáveis do daemon de script em um outro servidor.

Durante a instalação do IBM InfoSphere Federation Server, os arquivos executáveis do daemon de script são instalados no servidor federado. O nome e a localização do arquivo são os seguintes:

UNIX db2script_daemon está instalado no diretório \$DB2PATH/bin. \$DB2PATH é o diretório no qual o servidor federado está instalado.

Windows

db2script_daemon.exe está instalado no diretório %DB2PATH%\bin. %DB2PATH% é o diretório no qual o servidor federado está instalado, normalmente C:\SQLLIB\bin.

Se for utilizado um servidor de script separado, copie os arquivos executáveis e de configuração do daemon de script do servidor federado para o servidor de script. Os arquivos executáveis do daemon de script podem ser executados em qualquer diretório no servidor de script que não contém espaços nos nomes no caminho do diretório.

2. Certifique-se de que o arquivo de configuração para o daemon de script esteja no servidor correto.

Durante a instalação do sistema federado, um arquivo de configuração de amostra para o daemon de script é instalado no servidor federado. O nome do arquivo de configuração de amostra é `SCRIPT_DAEMON.config`. A localização do arquivo é a seguinte:

UNIX O arquivo de configuração do daemon é instalado no diretório `$DB2PATH/bin`.

Windows

O arquivo de configuração do daemon está instalado no diretório `%DB2PATH%\bin`.

Por padrão, o daemon procura pelo arquivo de configuração no diretório de trabalho a partir do qual o daemon é iniciado. Você pode copiar o arquivo de configuração para outro local. Se um servidor de script for utilizado, copie o arquivo de configuração do daemon a partir do diretório no servidor federado para um diretório no servidor de script. É possível copiar o arquivo de configuração do daemon para qualquer diretório no servidor de script que o daemon pode acessar.

3. Edite o arquivo de configuração de amostra para o daemon de script.
 - a. Renomeie o arquivo de configuração para que possa ser possível utilizar o arquivo de amostra novamente.
 - b. Certifique-se de que a primeira linha no arquivo de configuração seja um sinal de igual (=). Se o sinal de igual estiver faltando, o daemon não inicia. Uma mensagem de erro indicará que o `DAEMON_PORT` não foi especificado.
 - c. Verifique se a última linha do arquivo de configuração termina com uma nova linha.

A arquivo de configuração de amostra que é fornecido com o sistema federado termina com um caractere de nova linha. Se a última linha não terminar com um caractere de nova linha, você receberá uma mensagem de erro quando tentar executar sua primeira consulta de script que utiliza a origem de dados listada na última linha.
 - d. Certifique-se de que não haja espaços extras depois dos caminhos de diretório ou no final do arquivo de configuração.
 - e. Especifique as opções abaixo no arquivo de configuração. Para opções que necessitam de caminho, é possível especificar caminhos relativos. Caminhos relativos são relativos ao diretório a partir do qual o processo do daemon foi iniciado.

DAEMON_PORT=*port_number*

A porta da rede na qual o daemon atende pedidos de tarefa de script enviados pelo wrapper. O valor padrão é 4099.

MAX_PENDING_REQUESTS=*number_of_requests*

O número máximo de pedidos de tarefa de script que pode ser bloqueado no daemon num mesmo momento. Este número não representa o número de tarefas de script que estão em execução no momento, somente o número de pedidos de tarefas que podem ser bloqueados no mesmo momento. Configure este valor para um número maior do que cinco. O daemon de script não restringe o número de tarefas de scripts que podem ser executadas simultaneamente.

DAEMON_LOGFILE_DIR=*dir*

O diretório no qual o daemon cria seu arquivo de log. Este arquivo contém informações de status e de erro que são geradas pelo daemon de script.

SCRIPT_OUT_DIR_PATH=*path*

O diretório no qual o daemon cria o arquivo temporário para armazenar os dados de saída do script. O daemon lê os dados a partir deste arquivo e passa os dados de volta para o wrapper através da conexão de rede. Depois dos dados serem passados para o wrapper, o daemon apaga o arquivo temporário

script specification entry=*entry*

Uma lista de entradas que especificam o nome e a localização dos scripts que podem ser chamados pelo wrapper de script. A entrada tem este formato:

script_name=fully-qualified_script_path

Os exemplos a seguir aplicam-se ao sistema operacional designado:

UNIX Por exemplo, para especificar um script que acessa uma origem de dados Oracle, inclua a linha a seguir ao arquivo de configuração do daemon:

oracle=/dsk/1/data/oracle

Windows

Por exemplo, para especificar um script que acessa uma origem de dados Oracle, inclua a linha a seguir ao arquivo de configuração do daemon:

oracle=c:\data\oracle.a

Exemplo

O exemplo a seguir mostra um arquivo SCRIPT_DAEMON.cfg para quatro scripts:

```
=
DAEMON_PORT=4099
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=./
SCRIPT_OUT_DIR_PATH=./
fee=/home/user_id/fee
fie=/home/user_id/fie
foe=/home/user_id/foe
fum=/home/user_id/fum
```

O arquivo de configuração de amostra para o daemon de script fornece um exemplo de configuração do daemon de script.

Iniciando o Daemon de Scripts

Antes de poder acessar as origens de dados com scripts, inicie o daemon de scripts.

Antes de Iniciar

Você precisa ter acesso de gravação a todos os caminhos listados no arquivo de configuração do daemon para as opções DAEMON_LOGFILE_DIR e SCRIPT_OUT_DIR_PATH.

Sobre Esta Tarefa

O arquivo executável para o daemon de scripts inicia um novo processo no qual o daemon de scripts é executado.

Procedimento

1. Abra o diretório no qual o arquivo executável do daemon está localizado.
2. Emita o comando **db2script_daemon** para executar os arquivos executáveis, incluindo as opções adequadas.

Exemplo

O exemplo a seguir inclui as opções:

```
db2script_daemon -a action -c config_file -d debug_level -u user_id -p password
```

Comando db2script_daemon - Opções e Exemplos:

O comando db2script_daemon inicia o daemon de scripts. É possível iniciar o daemon de scripts com diversas opções.

O comando **db2script_daemon** pode ser utilizado em servidores UNIX ou Windows. Algumas das opções listadas na sintaxe só podem ser usadas em servidores Windows.

As opções especificadas com a ação start afetam somente a instância atual do daemon e sobrepõem os valores especificados com a ação install.

Opções - Comando db2script_daemon

O comando db2script_daemon possui as seguintes opções:

-a *ação* (somente para Windows)

Executa a atividade especificada. As ações válidas são status, install, start, stop e remove.

-c *config_file*

Instrui o serviço daemon a utilizar o arquivo de configuração especificado. Se o arquivo de configuração não for especificado, o daemon procura pelo arquivo SCRIPT_DAEMON.config no diretório onde os arquivos executáveis do daemon estão instalados. É possível utilizar esta opção com as ações install e start.

-d [1 | 2 | 3]

Configura o nível de depuração para o serviço daemon com o valor especificado. Um valor 1 ativa a criação de log, 2 rastreia todos os comandos e 3 ativa a criação de log, rastreia todos os comandos e salva todos os arquivos temporários para capturar a saída XML. É possível utilizar esta opção com as ações install e start.

-u *user_id* (somente para Windows)

Configura o serviço daemon para executar sob o ID de usuário especificado. Utilize esta opção com a ação install.

-p *senha* (somente para Windows)

Especifica a senha do ID de usuário especificado. A senha só é válida e necessária quando você especifica a opção -u. Se a opção -p não for especificada quando você definir a opção -u, o programa solicitará a senha. Utilize esta opção com a ação install.

Exemplos - Comando db2script_daemon

Os exemplos a seguir mostram como utilizar as opções do daemon de scripts.

Inicie o daemon

Para iniciar o daemon no UNIX, emita o seguinte comando:

```
db2script_daemon
```

Esse comando presume que o arquivo de configuração do daemon esteja no mesmo diretório do arquivo executável.

Para instalar e iniciar o daemon no Windows, emita os seguintes comandos:

```
db2script_daemon -a install  
db2script_daemon -a start
```

Especificar o arquivo de configuração do daemon

Se o nome do arquivo de configuração do daemon foi alterado ou se o arquivo de configuração não estiver no mesmo diretório que o arquivo executável do daemon, utilize a opção -c quando executar o arquivo executável. Esta opção especifica o caminho do diretório e o nome para o arquivo de configuração do daemon.

Neste exemplo, as informações de configuração do daemon encontram-se em um arquivo chamado SCRIPT_D.config no subdiretório cfg em um servidor UNIX. Emita o seguinte comando:

```
db2script_daemon -c cfg/SCRIPT_D.config
```

Especifique o nível de depuração

Se desejar iniciar o daemon com a depuração ativada com um nível de depuração 2, emita os seguintes comandos:

```
db2blast_daemon -a install -d 2  
db2blast_daemon -a start
```

Verifique o status do daemon (Windows)

Para verificar o status do daemon em um servidor Windows, execute o seguinte comando:

```
db2blast_daemon -a status
```

Esse comando presume que o arquivo de configuração do daemon esteja no mesmo diretório do arquivo executável.

Interromper o daemon

Para parar o daemon no UNIX, liste o ID do processo do daemon emitindo o seguinte comando:

```
ps -ef | grep db2script
```

Então utilize o ID do processo para parar o daemon emitindo o seguinte comando:

```
kill process_ID
```

Este comando assume que o arquivo de configuração do daemon está no mesmo diretório do arquivo executável.

Para interromper o daemon no Windows, execute o seguinte comando:

```
db2script_daemon -a stop
```

Remova o daemon

É possível remover o daemon de scripts quando não desejar mais utilizar o wrapper do script.

Para remover o daemon, execute este comando:

```
db2script_daemon -a remove
```

Registrando o Wrapper de Script

Registre o wrapper de script para acessar as origens de dados com os scripts. O wrapper de script é implementado como um arquivo de biblioteca.

Procedimento

Emita a instrução CREATE WRAPPER, especificando um nome para o wrapper de script e o nome do arquivo da biblioteca do wrapper.

Por exemplo, para registrar um wrapper com o nome script_wrapper em um servidor federado que utiliza AIX, emita a seguinte instrução:

```
CREATE WRAPPER script_wrapper LIBRARY 'libdb2lsscript.a';
```

O nome do arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado.

Não há opções específicas de wrapper de script para a instrução CREATE WRAPPER do wrapper de script. Por padrão, o wrapper executa sem proteção.

Arquivo de Biblioteca do Wrapper de Script:

Para registrar o wrapper de script, especifique o arquivo de biblioteca do wrapper de script para o sistema operacional do servidor federado.

Quando você instala a federação, um arquivo de biblioteca do wrapper de script é incluído ao caminho do diretório padrão.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 34. Locais da Biblioteca do Wrapper de Script e Nomes de Arquivos

Sistema Operacional	Caminho do Diretório	Nome do arquivo da biblioteca do wrapper
AIX	/usr/opt/install_path/lib	libdb2lsscript.a
Linux	/opt/IBM/db2/install_path/lib	libdb2lsscript.so
Solaris	/opt/IBM/db2/install_path/lib	libdb2lsscript.so
Windows	%DB2PATH%\bin	db2lsscript.dll

- *install_path* é o caminho do diretório onde a federação é instalada no UNIX ou Linux.
- %DB2PATH% é a variável de ambiente que é utilizada para especificar o caminho do diretório onde a federação está instalada no Windows. O caminho de diretório padrão no Windows é C:\Program Files\IBM\SQLLIB.

Registrando a Definição do Servidor Para Um Script Como Uma Origem de Dados

Registre cada servidor que deseja acessar no banco de dados federado.

Procedimento

Emita a instrução CREATE SERVER.

Por exemplo:

```
CREATE SERVER script_server WRAPPER script_wrapper  
  OPTIONS (NODE 'myserver.example.com', DAEMON_PORT '4099');
```

As opções de servidor NODE e DAEMON_PORT são requeridas para scripts como origens de dados.

Após o registro da definição do servidor, utilize a instrução ALTER SERVER para incluir ou descartar opções do servidor.

Instrução CREATE SERVER - Exemplos para o Wrapper de Script:

Os exemplos mostram a utilização das opções requeridas e as opções de servidor adicionais.

Exemplo de opções requeridas

O exemplo a seguir mostra como registrar uma definição de servidor para o wrapper de script emitindo a instrução CREATE SERVER:

```
CREATE SERVER server1_scriptn WRAPPER script_wrapper  
  OPTIONS(NODE 'big_rs.company.com');
```

server1_scriptn

Um nome que você designa ao servidor de script. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER *script_wrapper*

O nome do wrapper.

NODE *'big_rs.company.com'*

O nome do host do sistema no qual o processo do daemon de script está sendo executado. Este valor faz distinção entre maiúsculas e minúsculas.

Apesar do nome do nó ser especificado como uma opção na instrução CREATE SERVER, ele é requerido para o wrapper de script.

Exemplo de opção opcional

O exemplo a seguir mostra uma opção de servidor adicional que pode ser especificada quando uma definição de servidor é registrada para o wrapper de script:

```
CREATE SERVER server1_scriptn WRAPPER script_wrapper  
  OPTIONS(NODE 'big_rs.company.com', DAEMON_PORT '4088');
```

DAEMON_PORT *'4088'*

Especifica o número da porta que o daemon atende para pedidos de tarefas de script. O número da porta deve ser o mesmo número especificado na opção DAEMON_PORT do arquivo de configuração do daemon. O número da porta padrão é 4099.

Registrando apelidos para scripts

Registre um apelido separado para cada script. Utilize esses apelidos quando consultar a origem de dados que é acessada por um script.

Sobre Esta Tarefa

O wrapper de script associa dados XML a apelidos. Apelidos pai e filho correspondem aos elementos raiz e aninhados em um documento XML. Os apelidos pai e filho estão conectados por chaves primárias e estrangeiras que são especificadas na instrução CREATE NICKNAME. Cada apelido é definido por expressões XPath que identificam os elementos de dados XML e especificam como extrair valores de coluna de cada elemento.

A origem de dados é especificada pela instrução CREATE NICKNAME e é associada a um nome de script com a opção de apelido DATASOURCE. Crie uma coluna para cada argumento de entrada a ser passado para o script. Utilize as opções de coluna de entrada para controlar a sintaxe para chaves nas linhas de comandos. O valor para cada chave deve ser incluído no predicado de consulta pela utilização da função customizada ARGS em tempo de execução.

Scripts simples que não exigem argumentos de linha de comandos não precisam de colunas de entrada.

Procedimento

Emita a instrução CRIAR APELIDO a partir da linha de comandos.

Por Exemplo:

```
CREATE NICKNAME nickname
(
  column_name data_type OPTIONS ('nickname_column_options'),
  column_name data_type OPTIONS ('nickname_column_options'),
  column_name data_type OPTIONS ('nickname_column_options')
)
FOR SERVER server_definition_name
OPTIONS (nickname_options);
```

Apelidos podem ter até 128 caracteres de comprimento.

O que Fazer Depois

Emita a instrução para cada script para o qual desejar criar um apelido.

Instrução CREATE NICKNAME - Exemplos para o Wrapper de Script

Os exemplos mostram como utilizar a instrução CREATE NICKNAME para registrar apelidos para o wrapper de script.

O exemplo a seguir cria um apelido pai para os dados XML que são retornados de um script chamado fee:

```
CREATE NICKNAME customers
(
  argle double OPTIONS(SWITCH '-argle', POSITION 1, DEFAULT 1.0 ),
  argfile CLOB() OPTIONS(SWITCH '-file', INPUT_MODE 'FILE_INPUT', POSITION 2),
  argpos varchar() OPTIONS(SWITCH ' ', POSITION 3),
  id VARCHAR(5) OPTIONS(XPATH './@id'),
  name VARCHAR(16) OPTIONS(XPATH './name'),
  address VARCHAR(30) OPTIONS(XPATH './address/@street'),
  cid VARCHAR(16) FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER script_server
OPTIONS(DATASOURCE 'fee',
  XPATH '/doc/customer', STREAMING 'YES');
```

O exemplo a seguir cria um apelido chamado orders. O apelido orders é um filho do apelido chamado customers, que foi criado no exemplo anterior:

```
CREATE NICKNAME orders
(
    amount INTEGER          OPTIONS(XPATH './amount'),
    date   VARCHAR(10)     OPTIONS(XPATH './date'),
    oid    VARCHAR(16)     OPTIONS(PRIMARY_KEY 'YES'),
    cid    VARCHAR(16) FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
FOR SERVER script_server
OPTIONS( XPATH './order');
```

Opções de Apelido do Wrapper de Script

É possível especificar opções ao se criar um apelido para um script. Somente o apelido raiz pode incluir colunas de entrada.

Opções de Apelido

A lista a seguir descreve as opções de apelidos:

DATASOURCE

O nome do script que será chamado. O script deve ser listado no arquivo de configuração do daemon do script. Esta opção é requerida para dados de origem no apelido pai. Esta opção aplica-se somente ao apelido raiz.

NAMESPACES

Uma lista separada por vírgulas de pares de nome=valor que o wrapper utiliza para solucionar prefixos de espaço de nomes na expressão XPath do apelido.

TIMEOUT

O tempo máximo, em minutos, que o wrapper de script espera por resultados a partir do daemon de script. O valor padrão é de 60 minutos. Esta opção aplica-se somente ao apelido raiz.

VALIDATE

Especifica se um documento de origem XML deve ser validado antes dos dados XML serem extraídos. Se esta opção for configurada como YES, o sistema de banco de dados do DB2 verifica se a estrutura do documento de origem está de acordo com um esquema XML ou com uma DTD (Document Type Definition). Esta opção é aceita apenas para colunas do apelido raiz (o apelido que identifica os elementos no nível mais elevado do documento XML). O valor padrão é NO.

O documento de origem XML não é validado se o wrapper de script não puder localizar o arquivo de esquema XML ou arquivo DTD (.xsd ou .dtd). O sistema de banco de dados do DB2 não emite uma mensagem de erro se a validação não ocorrer. Certifique-se de que o arquivo de esquema XML ou arquivo DTD está no local que foi especificado no documento de origem XML. Não configure o parâmetro VALIDATE como YES se configurar o parâmetro STREAMING como YES.

STREAMING

Especifica se o documento de origem XML está separado em fragmentos lógicos que correspondem ao nó que é compatível com a expressão XPath do apelido. O wrapper então processa os dados de origem XML fragmento por fragmento, reduzindo a utilização da memória total. Esta opção é aceita apenas para colunas do apelido raiz (o apelido que identifica os elementos no nível mais elevado do documento XML). O valor padrão de STREAMING é NO. Não configure o parâmetro STREAMING como YES se configurar o parâmetro VALIDATE como YES.

XPATH

Uma expressão XPath que identifica o elemento XML que representa tuplas individuais na origem de dados. O wrapper avalia a opção do apelido XPATH para um apelido filho no contexto do caminho que é especificado pela opção do apelido XPATH do apelido pai. Esta expressão XPath é utilizada como um contexto para avaliação de valores de coluna que são identificados pelas opções de coluna do apelido XPATH.

Opções de Colunas de Apelidos

As opções de coluna do apelido são descritas no seguinte:

DEFAULT

O valor padrão para uma coluna de entrada. Essa opção se aplica apenas a colunas de entrada.

O valor padrão é utilizado se nenhum valor for fornecido pela consulta SQL. Essa opção não é necessária.

FOREIGN_KEY

Indica que este apelido é um apelido filho e especifica o apelido pai correspondente.

Um apelido pode ter no máximo uma opção de coluna FOREIGN_KEY. O valor para esta opção faz distinção entre maiúsculas e minúsculas. A coluna que é designada pela opção FOREIGN_KEY contém uma chave que é gerada pelo wrapper. O valor da coluna não pode ser recuperado em uma consulta SELECT, e a opção XPATH não deve ser especificada. A coluna somente pode ser utilizada para unir apelidos pais e apelidos filhos. Uma instrução CREATE NICKNAME com uma opção FOREIGN_KEY falha se o apelido pai tiver um nome de esquema diferente. A menos que o apelido referido em uma cláusula FOREIGN_KEY tenha sido explicitamente definido como minúsculo ou composto por letras maiúsculas e minúsculas demarcando isto com aspas na instrução CREATE NICKNAME correspondente, especifique o apelido em maiúscula ao se referir a esse apelido na cláusula FOREIGN_KEY.

As colunas da chave estrangeira devem ser designadas como FOR BIT DATA e NOT NULL.

INPUT_MODE

Especifica o modo de entrada para uma coluna. Os valores válidos são CONFIG ou FILE_INPUT. O wrapper transmite o valor especificado para o daemon do script.

CONFIG

O valor é tratado como um parâmetro configurável.

FILE_INPUT

É criado um arquivo que armazena o valor e o nome do arquivo é passado como um argumento da linha de comandos.

POSITION

Um valor inteiro para parâmetros posicionais. A ordem de seqüência da posição começa em 1. Esta opção aplica-se somente às colunas de entrada.

Se o valor posicional estiver configurado como um inteiro, essa entrada deverá estar nessa posição na linha de comandos. Se essa opção estiver configurada, a chave será inserida no local apropriado quando a consulta for executada. Se POSITION estiver configurado como -1, a opção será incluída como a última opção da linha de comandos. Por exemplo, se um valor de coluna tiver que estar no final de uma linha de comandos e não houver a opção SWITCH, a

configuração do valor de POSITION como -1 inclui o valor no final da linha de comandos. Os valores do inteiro POSITION não podem ser duplicados em um apelido. Essa opção não é necessária.

PRIMARY_KEY

Indica que esse apelido é um apelido pai. O tipo de dados da coluna deve ser VARCHAR(16). Um apelido pode ter no máximo uma opção de coluna PRIMARY_KEY. O único valor válido é Y.

A coluna que é designada como PRIMARY_KEY contém uma chave que é gerada pelo wrapper. O valor da coluna não pode ser recuperado em uma consulta SELECT, e a opção XPATH não deve ser especificada. A coluna somente pode ser utilizada para unir apelidos pais e apelidos filhos. As colunas da chave estrangeira devem ser designadas como FOR BIT DATA e NOT NULL.

SWITCH

Uma cadeia de caracteres para especificar um parâmetro para o script na linha de comandos. Essa opção se aplica apenas a colunas de entrada.

Na linha de comandos, o valor desta opção precede o valor da coluna fornecido por WSSCRIPT.ARGS ou pelo valor padrão, se houver. Se o valor para SWITCH for uma cadeia vazia e existir um valor padrão para a coluna, o valor padrão será incluído sem qualquer informação para SWITCH quando a linha de comandos for gerada. Se nenhum valor padrão for fornecido e nenhum valor para a coluna for fornecido pela consulta SQL, então esta coluna de entrada será ignorada quando a linha de comandos for gerada. Esta opção é necessária para uma coluna de entrada.

SWITCH_ONLY

Permite o uso de chaves sem um argumento da linha de comandos.

Se a opção SWITCH_ONLY for especificada com um valor Y, os valores de saída válidos serão Y ou N. Para um valor de entrada Y, apenas a chave é incluída na linha de comandos. Para um valor de entrada N, nenhum valor é incluído na linha de comandos.

VALID_VALUES

Um conjunto de valores válidos separados por ponto-e-vírgula para uma coluna.

XPATH

Especifica a expressão XPath no documento XML que contém os dados que correspondem a essa coluna. O wrapper de script avalia a expressão XPath depois da instrução CREATE NICKNAME aplicar esta expressão XPath a partir desta opção de apelido XPATH. Se você executar uma consulta em um nome de coluna que tenha uma referência de tag XPATH configurada incorretamente como maiúsculas e minúsculas incorretas, sua consulta retorna valores nulos nesta coluna para todas as linhas retornadas.

Consultas SQL com o Wrapper de Script

Consultas SQL que são feitas através do wrapper de script utilizam a função customizada para transportar os valores de entrada do parâmetro para o script.

Toda instrução SELECT que passa valores de parâmetros para um script através do wrapper de script deve conter no mínimo um predicado com uma função customizada para aproveitar os valores de entrada para o script.

Apelidos Raiz

Por exemplo, a instrução a seguir cria um apelido raiz para um script chamado `myscript`:

```
CREATE NICKNAME customers (  
  argle double OPTIONS(SWITCH '-argle', POSITION 1, DEFAULT 1.0),  
  argfile CLOB() OPTIONS(SWITCH '-file', INPUT_MODE 'FILE_INPUT', POSITION 2),  
  argpos varchar() OPTIONS(SWITCH ' ', POSITION 3),  
  id varchar(10) OPTIONS(XPATH './@id'),  
  name varchar OPTIONS(XPATH '/name'))  
FOR SERVER script_server  
  OPTIONS(DATASOURCE 'myscript', XPATH 'doc/customer',  
    TIMEOUT '300', VALIDATE 'YES');
```

As instruções da função customizada são as seguintes:

```
CREATE FUNCTION wsscript.args (varchar(), varchar())  
  RETURNS INTEGER AS TEMPLATE  
  DETERMINISTIC NO EXTERNAL ACTION;
```

```
CREATE FUNCTION wsscript.args (date(), date())  
  RETURNS INTEGER AS TEMPLATE  
  DETERMINISTIC NO EXTERNAL ACTION;
```

```
CREATE FUNCTION wsscript.args (integer(), integer())  
  RETURNS INTEGER AS TEMPLATE  
  DETERMINISTIC NO EXTERNAL ACTION;
```

```
CREATE FUNCTION wsscript.args (CLOB(), CLOB())  
  RETURNS INTEGER AS TEMPLATE  
  DETERMINISTIC NO EXTERNAL ACTION;
```

```
CREATE FUNCTION wsscript.args (double(), double())  
  RETURNS INTEGER AS TEMPLATE  
  DETERMINISTIC NO EXTERNAL ACTION;
```

O arquivo de configuração especifica os seguintes parâmetros de configuração:

```
SCRIPT_OUT_DIR_PATH=C:\temp  
myscript=C:\perl\bin\perl myscript.pl -model
```

Para executar uma consulta e enviar o conteúdo para a tabela `t1.bigdata` para o `daemon` e para o arquivo local `C:\temp\f12345`, emita a seguinte consulta e o comando `wsscript.args`:

```
SELECT id, name FROM customers, t1  
  WHERE wsscript.args (customers.argfile, t1.bigdata) = 1
```

A consulta precedente resulta na seguinte linha de comandos:

```
C:\perl\bin\perl myscript.pl -model -argle 1.0 -file C:\temp\f12345
```

Para executar uma consulta que utiliza os valores padrão para todos os parâmetros, execute uma consulta no apelido sem predicados. Para evitar problemas com saída excessiva, inclua a opção `STREAMING` no apelido.

Apelidos Filhos

O wrapper de script mapeia o conjunto de resultados XML do script em apelidos que possuem uma relacionamento pai-filho. Para recuperar dados de um apelido filho, use o apelido filho ao seu apelido pai até a raiz. As instruções `SELECT` que fazem referência a um apelido filho devem ser unidas com o apelido pai do apelido filho utilizando as colunas-chave primárias e estrangeiras.

A consulta a seguir exibe os nomes dos clientes e quantidades para cada ordem de cada cliente:

```
SELECT c.name, o.amount FROM customers c, orders o
WHERE c.cid=o.cid
AND wsscript.args (customers.argfile, t1.bigdata) = 1
AND wsscript.args (customers.argpos, VARCHAR('test1')) = 1
AND wsscript.args (customers.argle, FLOAT('3.5')) = 1
```

Especifica a instrução de junção `c.cid=o.cid` para indicar o relacionamento pai-filho entre o apelido `customers` e o apelido `orders`. Se for feita uma junção de um apelido filho a si próprio, uma mensagem de erro será retornada.

Otimizando o Desempenho do Wrapper de Script

A localização do daemon de script pode afetar o desempenho da consulta.

Para aumentar o desempenho da comunicação de rede, utilize um servidor de script separado para o daemon de script. Coloque o servidor federado e o servidor de script em servidores separados. Coloque também o daemon de script no servidor de script.

Configurando o Acesso às Origens de Dados do Sybase

Para configurar um servidor federado para acessar origens de dados do Sybase você deve fornecer ao servidor federado informações sobre as origens de dados e os objetos que deseja acessar.

Antes de Iniciar

- O software cliente Sybase deve ser instalado no servidor que agirá como servidor federado. Ao instalar o cliente do Sybase em Windows, é preciso especificar a opção **Completo** ou **Personalizado**. Se a opção customizada foi especificada, é preciso a opção **XA Interface Library for ASE Distributed Transaction Manager**.
- IBM InfoSphere Federation Server deve estar instalado em um servidor que atue como servidor federado.
- Verifique a configuração do servidor federado.
- Verifique o parâmetro federado para assegurar que a federação está ativada.

Procedimento

1. Configure as variáveis de ambiente do Sybase.
2. Configure e teste o arquivo de configuração do cliente Sybase usando um dos seguintes métodos, dependendo de seu sistema operacional:
 - Configuração e teste do arquivo de configuração do cliente Sybase (Linux, UNIX).
 - Configuração e teste do arquivo de configuração do cliente (Windows).
3. Registre o wrapper.
4. Registre a definição do servidor.
5. Crie os mapeamentos de usuário.
6. Teste a conexão com o servidor Sybase.
7. Registre os apelidos para tabelas e visualizações do Sybase.

Suporte do Wrapper do Sybase para ASE (Adaptive Server Enterprise)

O wrapper do Sybase suporta o Sybase ASE (Adaptive Server Enterprise) 15.0, além do ASE 12.5 e do ASE 12.0.

Clientes Suportados pelo Sybase ASE 15.0

Não é possível se conectar ao ASE 15.0 usando o wrapper do Sybase com o cliente Sybase, versão 12.5.1 ou posterior

Se utilizar o SDK (Software Developer Kit), versão 12.5.1 como um cliente Sybase, a IBM recomenda que você instale o ESD (Electronic Software Distribution) nº12 ou posterior a partir do SDK no servidor federado com o wrapper do Sybase. Se utilizar o SDK, versão 15.0 como um cliente Sybase, a IBM recomenda que você instale o ESD nº3 ou posterior. Se não instalar o ESD, você pode receber um erro inesperado ao usar o Sybase.

Atualizando as Bibliotecas Sybase do Cliente Sybase, Versão 15.0

UNIX: Se utilizar o cliente Sybase, versão 15.0, é possível executar o script do Sybase **lnsyblibs** para atualizar os nomes da biblioteca Sybase para preservar a consistência entre as versões suportadas dos arquivos de biblioteca Sybase. O script **lnsyblibs** cria link simbólicos a partir de novos nomes de bibliotecas para os antigos nomes de bibliotecas que permitem que os aplicativos da pré-versão 15.0 funcionem com bibliotecas renomeadas.

Windows: Se utilizar o cliente Sybase, versão 15.0, é possível executar o arquivo **copylibs.bat** para copiar os arquivos *.dll necessários, o que permite que os aplicativos da pré-versão 15.0 funcionem com bibliotecas renomeadas.

Erro do Script lntsyblibs

O atual script **lnsyblibs** contém um problema. Ao executar 'lnsyblibs create', ocorre a seguinte mensagem de erro:

```
"libsy*.s[o: No such file or directory"
```

Como uma solução alternativa a esse problema, é possível excluir | (caracter de barra) entre [] (colchetes) na linha 34 do script. A Sybase está ciente do problema. Para obter mais informações, vá para o Suporte Sybase.

Suporte para junções de mesclagem

Junções de mesclagem do Sybase não são ativadas por padrão. Você deve ativar junções de mesclagem em ambos os servidores Sybase e federado antes de que possa usar junções de mesclagem com origens de dados do Sybase ASE:

1. Ative junções de mesclagem em seu servidor Sybase, consulte Enabling and disabling merge joins.
2. Execute a instrução ALTER SERVER para ativar junções de mesclagem em seu servidor federado:

```
ALTER SERVER sybase_server OPTIONS (ADD DB2_merge_join 'Y');
```

em que *sybase_server* é o nome do seu servidor Sybase.

Tipos de Dados Não-suportados

Não é possível criar apelidos para objetos de origem de dados que contenham tipos de dados não suportados. O wrapper do Sybase no suporta os seguintes tipos de dados introduzidos no ASE, versão 12.5.1:

- DATE

- TIME

O wrapper do Sybase no suporta os seguintes tipos de dados introduzidos no ASE, versão 15.0:

- BIGINT
- LONGSYSNAME
- UNITEXT
- UNSIGNED BIGINT
- UNSIGNED INT
- UNSIGNED SMALLINT

Definindo as Variáveis de Ambiente do Sybase

As variáveis de ambiente do Sybase devem ser configuradas no arquivo `db2dj.ini` no servidor federado.

Sobre Esta Tarefa

Restrições

Revise as restrições ao arquivo `db2dj.ini`.

O arquivo `db2dj.ini` contém informações de configuração sobre o software Sybase Open Client que está instalado no servidor federado.

Existem variáveis de ambiente requeridas e opcionais para origens de dados do Sybase.

Se você instalou o software Sybase Open Client antes de instalar o wrapper do Sybase, as variáveis de ambiente requeridas do Sybase estarão configuradas no arquivo `db2dj.ini`.

Você deve definir as variáveis de ambiente utilizando as etapas nesta tarefa caso não tenha instalado o software Sybase Open Client antes de instalar o wrapper do Sybase ou se desejar definir qualquer das variáveis de ambiente opcionais.

Procedimento

1. Utilize um dos seguintes métodos:

Método	Descrição
Configurar automaticamente as variáveis de ambiente.	Execute o assistente de instalação do IBM InfoSphere Federation Server. Siga as instruções no assistente. Importante: Configure as variáveis de ambiente necessárias executando o assistente de instalação. As variáveis de ambiente opcionais devem ser definidas manualmente.
Configure as variáveis de ambiente com a ferramenta de configuração do wrapper.	Execute a ferramenta de configuração do wrapper para configurar as variáveis de ambiente necessárias e as opcionais. Siga as instruções em Executando a ferramenta de configuração do wrapper.

Método	Descrição
Configurar manualmente as variáveis de ambiente.	<p>Edite o arquivo db2dj.ini:</p> <p>O arquivo db2dj.ini está localizado no arquivo especificado pela variável de registro DB2_DJ_INI do DB2. Quando esta variável não estiver configurada, o padrão será o seguinte:</p> <p>Linux ou UNIX <i>instancehome</i>/sql1lib/cfg/db2dj.ini <i>instancehome</i> O diretório inicial do proprietário da instância.</p> <p>Windows %DB2PATH%\cfg\db2dj.ini %DB2PATH% O diretório em que o sistema de banco de dados do DB2 está instalado, por exemplo: C:\Program Files\IBM\sql1lib.</p> <p>Se o arquivo db2dj.ini não existir, é necessário criá-lo com um editor de texto especificando as seguintes variáveis de ambiente Sybase:</p> <p>SYBASE O caminho de diretório no qual o software Sybase Open Client está instalado, por exemplo: SYBASE=/sybase</p> <p>SYBASE_OCS O nome da variável de ambiente SYBASE_OCS do diretório OCS, por exemplo: SYBASE_OCS=OCS-12_5</p> <p>Importante: Erros ocorrem se as variáveis de ambiente Sybase não estiverem especificadas no arquivo db2dj.ini.</p>

- Em Linux e UNIX, é necessário atualizar o arquivo .profile que se encontra na instância de banco de dados federado.

Execute os seguintes comandos para atualizar o arquivo .profile com as variáveis de ambiente Sybase que você incluiu no arquivo db2dj.ini:

```
export SYBASE=sybase_home_directory
export SYBASE_OCS=OCS-version_release
export PATH=${SYBASE}/bin:$PATH
```

sybase_home_directory

O diretório no qual o software Sybase Open Client está instalado.

sybase_ocs_directory

O nome do diretório OCS e é requerido.

- A partir do diretório inicial, execute o arquivo .profile que se encontra na instância de banco de dados federado para iniciar a instância de banco de dados DB2:

```
. .profile
```

- Em alguns sistemas operacionais, como o Linux, é necessário adicionar o caminho da biblioteca de cliente Sybase na variável DB2LIBPATH db2set, por exemplo:

```
db2set DB2LIBPATH=/opt/sybase125/OCS-12_5/lib
```

5. Para assegurar que as variáveis de ambiente estejam configuradas no servidor federado, é necessário reciclar a instância do banco de dados federado parando e reiniciando-a com os seguintes comandos:

```
db2stop  
db2start
```

O que Fazer Depois

Após concluir essa tarefa, você deverá registrar o wrapper.

Variáveis de Ambiente Sybase

Existem variáveis de ambiente requeridas e opcionais para origens de dados do Sybase. Essas variáveis são definidas no arquivo `db2dj.ini`.

As variáveis de ambiente a seguir são válidas para o Sybase:

- SYBASE
- SYBASE_OCS
- SYBASE_CHARSET (opcional)

Descrições da Variável

SYBASE

Especifica o caminho de diretório no qual o software Sybase Open Client está instalado. Especifique o caminho completo para essa variável de ambiente.

Por exemplo, se o Sybase Open Client Versão 12.5 estiver instalado no caminho do diretório `D:\djxc\lient\sybase\V125`, especifique a variável de ambiente SYBASE a seguir:

```
SYBASE=D:\djxc\lient\sybase\V125
```

Se o Sybase Open Client Versão 12.0 estiver instalado no caminho de diretório `D:\djxc\lient\sybase\V12`, especifique a variável de ambiente SYBASE a seguir:

```
SYBASE=D:\djxc\lient\sybase\V12
```

SYBASE_OCS

Especifica o diretório, versão e release do software Sybase Open Client instalado. Não especifique o caminho completo ao especificar essa variável de ambiente.

```
SYBASE_OCS=OCS-version_release
```

Por exemplo, se o Sybase Open Client Versão 12.0 estiver instalado no caminho de diretório `D:\djxc\lient\sybase\V12\OCS-12_0`, especifique o seguinte valor para a variável de ambiente SYBASE_OCS:

```
SYBASE_OCS=OCS-12_0
```

Se o Sybase Open Client Versão 12.5 estiver instalado no caminho de diretório `D:\djxc\lient\sybase\V125\OCS-12_5`, especifique o seguinte valor para a variável de ambiente SYBASE_OCS:

```
SYBASE_OCS=OCS-12_5
```

SYBASE_CHARSET

Especifica o nome do conjunto de caracteres que você deseja usar.

Configure a variável de ambiente SYBASE_CHARSET para o conjunto de

código que você especifica no parâmetro **CODESET** do servidor federado. Uma lista de nomes de conjuntos de caracteres válidos existe no diretório `$SYBASE\charsets`.

Os conjuntos de código são especificados diferentemente entre o parâmetro **CODESET** e a variável de ambiente `SYBASE_CHARSET`. Por exemplo, se você configurar o parâmetro **CODESET** para o Formato de Transformação de Unicode de 8 bits, UTF-8, então especifique UTF8 na variável de ambiente `SYBASE_CHARSET`:

```
SYBASE_CHARSET=utf8
```

Se a variável de ambiente `SYBASE_CHARSET` não for configurada, o wrapper usa o conjunto de caracteres do Sybase que corresponda ao especificado na página de códigos do banco de dados federado. Se não houver um conjunto de caracteres do Sybase correspondente, o wrapper usa o conjunto de caracteres `iso_1`.

Configurando e Testando o Arquivo de Configuração do Cliente Sybase (Windows)

O arquivo de configuração do cliente Sybase é utilizado para conectar-se a bancos de dados Sybase utilizando as bibliotecas do cliente instaladas no servidor federado.

Antes de Iniciar

O software cliente Sybase deve ser instalado no servidor federado.

Sobre Esta Tarefa

O arquivo de configuração do cliente especifica o local de cada Sybase SQL Server e instância do Adaptive Server Enterprise e o tipo de conexão (protocolo) para o servidor de banco de dados.

É preciso configurar um arquivo de configuração do cliente em cada instância no servidor federado usado para se conectar ao Sybase.

Procedimento

Para configurar e testar o arquivo de configuração do cliente Sybase nos servidores federados que executam o Windows:

Procedimento

1. Configure o arquivo de configuração do cliente usando o utilitário que vem com o software Sybase Open Client. Consulte a documentação para obter mais informação sobre o uso desse utilitário.

O arquivo de configuração do cliente é criado no diretório `%SYBASE%\ini`. O nome do arquivo é `sql.ini`.

2. Teste a conexão para se assegurar de que o software Sybase Open Client é capaz de se conectar ao servidor Sybase.

Use um utilitário de consulta Sybase apropriado, como `isql`, para testar a conexão.

Por exemplo, se o software Sybase Open Client estiver instalado no caminho do diretório `D:\djxc\lient\sybase\V125`, é possível emitir os seguintes comandos a partir do prompt de comandos:

```
cd D:\djxclient\sybase\V125\OCS-12_5\bin
isql -Ssybnode -Umary
```

Como alternativa, é possível emitir o seguinte comando a partir de um prompt de comandos:

```
%SYBASE%\%SYBASE_OCS%\bin\isql -Ssybnode -Umary
```

Especificando o caminho para o arquivo de interfaces

: Caso deseje utilizar um arquivo de interfaces diferente do arquivo padrão, use a opção do servidor IFILE para especificar o caminho. O wrapper do Sybase procura pelo arquivo de interfaces nos lugares a seguir, na ordem especificada:

- a. opção de servidor IFILE
- b. %DB2PATH%\interfaces
- c. %SYBASE%\ini\sql.ini

O que Fazer Depois

Depois de concluir essa tarefa, é possível configurar as variáveis de ambiente.

Configurando e Testando o Arquivo de Configuração do Cliente Sybase (UNIX)

O arquivo de configuração do cliente Sybase é utilizado para conectar-se a bancos de dados Sybase utilizando as bibliotecas do cliente instaladas no servidor federado.

Antes de Iniciar

O software cliente Sybase deve ser instalado no servidor federado.

Sobre Esta Tarefa

O arquivo de configuração do cliente especifica o local de cada Sybase SQL Server e instância do Adaptive Server Enterprise e o tipo de conexão (protocolo) para o servidor de banco de dados.

É preciso configurar um arquivo de configuração do cliente em cada instância no servidor federado usado para se conectar ao Sybase.

Procedimento

1. Configure o arquivo de configuração do cliente usando o utilitário que vem com o software Sybase Open Client.

O arquivo de configuração do cliente é criado no diretório \$SYBASE. O nome do arquivo padrão é interfaces. Consulte a documentação para obter mais informação sobre o uso desse utilitário.

2. Teste a conexão para se assegurar de que o software Sybase Open Client é capaz de se conectar ao servidor Sybase.

Use um utilitário de consulta Sybase apropriado, como isql, para testar a conexão.

Por exemplo, se o software Sybase Open Client estiver instalado no caminho do diretório /opt/djxclient/sybase/V125, é possível emitir os seguintes comandos a partir de um prompt UNIX:

```
cd /opt/djxclient/sybase/V125/OCS-12_5
isql -Ssybnode -Umary
```

Como alternativa, é possível emitir o seguinte comando a partir de um prompt UNIX:

```
$SYBASE/$SYBASE_OCS/bin/isql -Ssybnod -Umary
```

Especificando o caminho para o arquivo de interfaces

: Caso deseje utilizar um arquivo de interfaces diferente do arquivo padrão, use a opção do servidor IFILE para especificar o caminho. O wrapper do Sybase procura pelo arquivo de interfaces nos lugares a seguir, na ordem especificada:

- a. opção de servidor IFILE
- b. sqllib/interfaces
- c. \$SYBASE/interfaces

O que Fazer Depois

Depois de concluir essa tarefa, é possível configurar as variáveis de ambiente.

Registrando o Wrapper do Sybase

Você deve registrar um wrapper para acessar origens de dados Sybase. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Procedimento

Emita a instrução CREATE WRAPPER a partir da linha de comandos e especifique o nome padrão para o wrapper Sybase.

Por exemplo:

```
CREATE WRAPPER CTLIB
```

Lembre-se: Ao registrar um wrapper utilizando o nome padrão, CTLIB, o servidor federado utiliza automaticamente a biblioteca de wrapper apropriada do Sybase para o sistema operacional no qual o servidor federado está em execução. Se o nome padrão do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, você poderá substituir o nome padrão do wrapper por um de sua escolha. Se você não utilizar o nome padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.

Por exemplo, para registrar um wrapper com o nome de sybase_wrapper em um servidor federado que utiliza o AIX, emita a seguinte instrução:

```
CREATE WRAPPER sybase_wrapper  
LIBRARY 'libdb2ctlib.a';
```

O arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado.

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar a definição do servidor.

Arquivos de Biblioteca do Wrapper do Sybase

Os arquivos de biblioteca do wrapper do Sybase são incluídos no servidor federado ao instalar o wrapper.

Ao instalar o wrapper do Sybase, três arquivos de biblioteca são incluídos no caminho padrão do diretório. Por exemplo, se o servidor federado estiver em execução no AIX, os arquivos de biblioteca do wrapper incluídos no caminho do diretório são `libdb2ctlib.a`, `libdb2ctlibF.a` e `libdb2ctlibU.a`. Os outros arquivos de biblioteca do wrapper são usados internamente pelo wrapper do Sybase.

Se você não utilizar o nome padrão do wrapper ao registrar um wrapper, deverá incluir o parâmetro `LIBRARY` na instrução `CREATE WRAPPER` e especificar o nome do arquivo de biblioteca padrão.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 35. Locais de Biblioteca do Wrapper e Nomes de Arquivos do Sybase

Sistema Operacional	Caminho do Diretório	Nome do Arquivo de Biblioteca
AIX	<code>/usr/opt/install_path/lib32/</code> <code>/usr/opt/install_path/lib64/</code>	<code>libdb2ctlib.a</code>
Linux	<code>/opt/IBM/db2/install_path/lib32</code> <code>/opt/IBM/db2/install_path/lib64</code>	<code>libdb2ctlib.so</code>
Solaris	<code>/opt/IBM/db2/install_path/lib32</code> <code>/opt/IBM/db2/install_path/lib64</code>	<code>libdb2ctlib.so</code>
Windows	<code>%DB2PATH%\bin</code>	<code>db2ctlib.dll</code>

`install_path` é o caminho do diretório em que o IBM InfoSphere Federation Server está instalado no UNIX ou Linux.

Registrando Definições de Servidor para uma Origem de Dados Sybase

Você deverá registrar cada servidor Sybase que deseja acessar no banco de dados federado.

Procedimento

1. Localize o nome do nó no arquivo de interfaces do Sybase. OS exemplos a seguir mostram as entradas para os arquivos de interfaces para servidores federados que executam UNIX ou Windows:

UNIX:

```
sybase125
query tcp ether anaconda 4100
```

Windows:

```
[sybase125]
query=TCP,anaconda,4100
```

- A primeira linha em cada exemplo corresponde ao nome do nó, como `sybase125`.
 - A segunda linha lista o tipo de conexão, o nome do host e o número da porta. Neste exemplo, `TCP` indica que se trata de uma conexão TCP/IP, `anaconda` corresponde ao nome do host e `4100` corresponde ao número da porta.
2. Para criar a definição de servidor, emita a instrução `CREATE SERVER` a partir da linha de comandos.

Por exemplo:

```
CREATE SERVER server_definition_name TYPE SYBASE
    VERSION Sybase_client_version_number WRAPPER wrapper_name
    OPTIONS (NODE 'node_name', DBNAME 'database_name');
```

Embora as variáveis '*node_name*' e '*database_name*' sejam especificadas como opções na instrução CREATE SERVER, essas opções são requeridas para origens de dados de Sybase.

Importante: Se o arquivo `sql.ini` não for renomeado para interfaces durante a configuração do arquivo de configuração do cliente Sybase, é preciso incluir a opção do servidor IFILE ao registrar a definição do servidor.

Após o registro da definição do servidor, utilize a instrução ALTER SERVER para incluir ou descartar opções do servidor.

No IBM Data Studio, é possível inserir comandos no editor de SQL e XQuery e implementar os comandos. Para obter mais detalhes, consulte [AddTopicRef](#).

O que Fazer Depois

Após concluir esta tarefa, você pode criar mapeamentos de usuário.

Instrução CREATE SERVER - Exemplos para o Wrapper do Sybase

Utilize a Instrução CREATE SERVER para Registrar Definições de Servidor para o Wrapper do Sybase. Este tópico fornece um exemplo completo com os parâmetros requeridos e um exemplo com opções de servidor adicionais.

O Exemplo a Seguir Mostra como Registrar uma Definição de Servidor para um Wrapper do Sybase Utilizando a Instrução CREATE SERVER:

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
    OPTIONS (NODE 'sybnode', DBNAME 'sybdb');
```

SYBSERVER

Um nome que você atribui ao servidor Sybase. Nomes de definição de servidor duplicados não são permitidos.

TYPE SYBASE

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso. Para o wrapper do CTLIB, o tipo de servidor deve ser SYBASE.

VERSION 12.0

A versão do software cliente de banco de dados Sybase que está sendo usada para a conexão federada.

WRAPPER CTLIB

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

NODE '*sybnode*'

O nome do nó no qual o servidor Sybase reside. Obtenha o nome do nó a partir do arquivo interfaces. Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, é necessário para as origens de dados do Sybase.

DBNAME '*sybdb*'

O nome do banco de dados Sybase que deseja acessar. Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do banco de dados seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dados do Sybase.

Opções do Servidor

Quando você cria uma definição do servidor, é possível especificar opções adicionais do servidor na instrução CREATE SERVER. As opções do servidor podem ser gerais ou específicas do Sybase.

Opção do Servidor DB2_ID_QUOTE_CHAR

Para criar apelidos para as tabelas com nomes de colunas que tenham mais de 28 caracteres, você deverá especificar a opção do servidor DB2_ID_QUOTE_CHAR e configurá-la para um espaço em branco único entre aspas simples, por exemplo:

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
  OPTIONS (NODE 'sybnode', DBNAME 'sybdb',
    DB2_ID_QUOTE_CHAR ' ');
```

opção do servidor IFILE

Se você não estiver usando os arquivos de interface padrão, você deve criar um arquivo interfaces e incluir a opção do servidor IFILE na instrução CREATE SERVER. O arquivo de interface padrão é \$SYBASE/interfaces para Linux e UNIX e %SYBASE%\ini\sql.ini para Windows.

O valor especificado para a opção do servidor IFILE corresponde ao caminho completo e nome do arquivo sql.ini do Sybase Open Client.

O exemplo a seguir mostra como usar a opção do servidor IFILE ao registrar uma definição de servidor em um servidor federado que executa o Windows:

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
  OPTIONS (NODE 'sybnode', DBNAME 'sybdb',
    IFILE 'C:\Sybase\ini\sql.ini');
```

opção do servidor TIMEOUT

A opção do servidor TIMEOUT configura o número de segundos que o wrapper aguarda por uma resposta a partir do servidor Sybase. Use a opção TIMEOUT para evitar conflitos em transações.

O exemplo a seguir mostra como especificar a opção do servidor IFILE ao registrar uma definição de servidor:

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
  OPTIONS (NODE 'sybnode', DBNAME 'sybdb',
    TIMEOUT '60');
```

As opções adicionais do servidor específicas do Sybase são:

- LOGIN_TIMEOUT
- PACKET_SIZE

Criando os Mapeamentos de Usuário para uma Origem de Dados Sybase

Quando você tenta acessar um servidor Sybase, o servidor federado estabelece uma conexão com o servidor Sybase utilizando um ID de usuário e uma senha

válidos para a origem de dados. Você deve definir uma associação (um mapeamento de usuário) entre o ID de usuário e a senha de cada servidor federado e o ID de usuário e a senha correspondentes da origem de dados.

Sobre Esta Tarefa

Crie um mapeamento de usuário para cada ID de usuário que acessará o sistema federado para enviar pedidos distribuídos à origem de dados do Sybase.

Procedimento

Emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR local_userID SERVER server_definition_name
    OPTIONS (REMOTE_AUTHID 'remote_userID', REMOTE_PASSWORD 'remote_password');
```

O que Fazer Depois

Depois de concluir essa tarefa, teste a conexão com o servidor Sybase.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper do Sybase

Utilize a instrução CREATE USER MAPPING para mapear um ID de usuário do servidor federado para um ID de usuário e senha do servidor Sybase. Este tópico fornece um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o USER de registro especial do DB2 com a instrução CREATE USER MAPPING.

O exemplo a seguir mostra como mapear um ID de usuário do servidor federado para um ID de usuário e senha do servidor Sybase:

```
CREATE USER MAPPING FOR maria SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'mary', REMOTE_PASSWORD 'day2night');
```

maria Especifica o ID de usuário local que você está mapeando para um ID de usuário definido no servidor Sybase.

SERVER SYBSERVER

Especifica o nome da definição do servidor registrado na instrução CREATE SERVER para o servidor Sybase.

REMOTE_AUTHID '*mary*'

Especifica o ID de usuário no servidor Sybase para o qual você está mapeando *maria*. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_ID para 'U' ou 'L' na instrução CREATE SERVER.

Embora o ID do usuário remoto seja especificado como uma opção na instrução CREATE SERVER, é necessário para as origens de dados do Sybase.

REMOTE_PASSWORD '*day2night*'

Especifica a senha associada a '*mary*'. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

Embora a senha remota seja especificada como uma opção na instrução CREATE SERVER, é necessário para as origens de dados do Sybase.

USER de Registro Especial do DB2

Você pode utilizar o USER de registro especial do DB2 para mapear o ID de autorização da pessoa que está emitindo a instrução CREATE USER MAPPING para o ID de autorização da origem de dados especificado na opção de usuário REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'mary', REMOTE_PASSWORD 'day2night');
```

Testando a Conexão com o Servidor Sybase

Teste a conexão com o servidor de origem de dados do Sybase para determinar se o servidor federado está configurado adequadamente para acessar as origens de dados do Sybase.

Sobre Esta Tarefa

Você pode testar a conexão com o servidor do Sybase utilizando a definição do servidor e os mapeamentos de usuário que definiu.

Procedimento

Abra uma sessão de passagem e emita uma instrução SELECT nas tabelas do sistema Sybase. Se a instrução SELECT retornar uma contagem, a definição do servidor e o mapeamento de usuário estão configurados apropriadamente.

Por exemplo:

```
SET PASSTHRU server_definition_name
SELECT count(*) FROM dbo.sysobjects
SET PASSTHRU RESET
```

Se a instrução SELECT retornar um erro, você deverá resolver os problemas dos erros de conexão.

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar apelidos para tabelas e visualizações do Sybase.

Resolução de Problemas de Erros de Conexão da Origem de Dados

Uma conexão de teste com o servidor de origem de dados pode retornar um erro por diversas razões. Há ações que você pode executar para determinar a causa de um erro.

Sintoma

É retornado um erro quando você tentar se conectar à origem de dados.

Causa

Há diversas causas possíveis para um problema de conexão.

Solução do Problema

Para resolver problemas de erros de conexão com a origem de dados, verifique os seguintes itens:

- Verifique se a origem de dados está disponível.
- Se aplicável, certifique-se de que o servidor de origem de dados esteja configurado para conexões de chegada.
- Certifique-se de suas configurações de mapeamento de usuário para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com a origem de dados. Altere o mapeamento de usuário ou crie outro, conforme necessário.
- Se aplicável, certifique-se de que o software do cliente da origem de dados no servidor federado esteja instalado e configurado corretamente para conexão com a origem de dados.
- Para origens de dados ODBC, certifique-se de que o driver ODBC no servidor federado esteja instalado e configurado corretamente para conexão com o servidor de origem de dados ODBC. Em servidores federados que executam Windows, utilize a ferramenta Administrador de Origem de Dados ODBC para verificar o driver. Em servidores de dados que executam UNIX, consulte a documentação do fornecedor do cliente ODBC.
- Verifique se as configurações para as variáveis configuradas no servidor federado estão corretas para a origem de dados. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis no arquivo db2dj.ini e as variáveis do DB2 Profile Registry (db2set).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e a crie novamente.

Registrando Apelidos para Tabelas e Visualizações do Sybase

Para cada definição de servidor Sybase registrada, você deverá registrar um pseudônimo para cada tabela ou visualização que deseja acessar. Utilize esses pseudônimos ao invés dos nomes dos objetos de origem de dados quando consultar os servidores do Sybase.

Antes de Iniciar

Atualize as estatísticas na origem de dados do Sybase antes de registrar um pseudônimo. O banco de dados federado confia nas estatísticas do catálogo da origem de dados para otimizar o processamento da consulta. Utilize o comando da origem de dados equivalente ao comando DB2 **RUNSTATS** para atualizar as estatísticas da origem de dados.

Procedimento

Emita a instrução CREATE NICKNAME a partir da linha de comandos.

Por exemplo:

```
CREATE NICKNAME nickname
```

```
FOR server_definition_name."remote_schema"."remote.table" ;
```

Os apelidos podem ter até 128 caracteres de comprimento.

Ao criar o apelido, o servidor federado consulta o catálogo de origem de dados utilizando o apelido. Esta consulta testa a conexão com a tabela ou visualização da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

O que Fazer Depois

Repita esta etapa para cada tabela ou visualização do Sybase para as quais deseja criar um apelido.

Instrução CREATE NICKNAME - Exemplos para o Wrapper do Sybase

Utilize a instrução CREATE NICKNAME para registrar um apelido para uma tabela ou visualização do Sybase que deseja acessar. Este tópico inclui um exemplo completo com os parâmetros requeridos.

O exemplo a seguir mostra como registrar um apelido para a tabela ou visualização Sybase usando a instrução CREATE NICKNAME.

```
CREATE NICKNAME SYBSALES FOR SYBSERVER."vinnie"."europe";
```

SYBSALES

Um apelido exclusivo que é utilizado para identificar a tabela ou visualização do Sybase.

Importante: O apelido é um nome de duas partes -o esquema e o apelido. Se você omitir o esquema ao registrar o pseudônimo, o esquema do pseudônimo será o ID de autorização do usuário que está registrando o pseudônimo.

```
SYBSERVER."vinnie"."europe"
```

Um identificador de três partes para o objeto remoto:

- *SYBSERVER* é o nome de definição do servidor atribuído ao servidor de banco de dados do Sybase na instrução CREATE SERVER.
- *vinnie* é o ID de usuário do proprietário ao qual a tabela ou visualização pertence.
- *europe* é o nome da tabela ou visualização remota que você deseja acessar.

O servidor federado fecha os nomes dos esquemas e tabelas do Sybase para maiúsculas, a menos que você coloque os nomes entre aspas.

Resolução de Problemas da Configuração do Wrapper do Sybase

Problemas ao carregar a biblioteca de wrapper do Sybase

Ao criar o wrapper do Sybase, é possível encontrar erros relacionados à instalação do software Sybase Open Client que impede que a biblioteca de wrapper do Sybase seja carregada.

Sintoma

Ao criar o wrapper do Sybase, o seguinte erro de SQL é emitido:

```
SQL10013N A biblioteca especificada "db2ctlibF.dll"  
pode ser carregada.
```

Causa

A Sybase XA Interface Library for ASE Distributed Transaction Manager não foi instalada no sistema Windows com o software Sybase Open Client.

Solução do Problema

Reinstale o software Sybase Open Client no Windows e selecione a opção de instalação **Completa** ou **Customizada**. Se você selecionar a opção de instalação **Customizada**, especifique a opção **XA Interface Library for ASE Distributed Transaction Manager**.

Variável de Ambiente do SYBASE Ausente

Se o arquivo `db2dj.ini` não estiver no diretório correto ou se estiver ausente, é preciso encontrar um erro de SQL.

Sintoma

O erro de SQL a seguir é emitido:

SQL1822N Código de erro inesperado "" recebido da origem de dados "server name". O texto e os tokens associados são "variável SYBASE não configurada".

Causa

O arquivo `db2dj.ini` não foi localizado ou o arquivo não contém a variável de ambiente SYBASE. O arquivo `db2dj.ini` está localizado no diretório especificado pela variável de registro `DB2_DJ_INI` do DB2. Quando a variável `DB2_DJ_INI` não está configurada, o arquivo `db2dj.ini` corresponde a um dos seguintes caminhos padrão, dependendo do sistema operacional:

- Em UNIX: `instancehome/sql1lib/cfg/db2dj.ini`.
instancehome
O diretório inicial do proprietário da instância.
- Em Windows: `%DB2PATH%\cfg\db2dj.ini`
`%DB2PATH%`
O diretório em que o sistema de banco de dados do DB2 está instalado, por exemplo, `C:\Program Files\IBM\sql1lib`.

Solução do Problema

Crie um arquivo `db2dj.ini` com as variáveis de ambiente Sybase necessárias e coloque-o no diretório correto para seu sistema operacional. Você pode usar um editor de texto para criar o arquivo `db2dj.ini`.

Nome do Nó do Sybase Ausente

Se o arquivo de configuração do cliente Sybase não for configurado adequadamente, o sistema federado pode não ser capaz de localizar o nome do nó Sybase.

Sintoma

O erro de SQL a seguir é emitido:

SQL1097N The node name was not found in the node directory.

Causa

O nome do nó do Sybase não foi localizado.

Solução do Problema

Para solucionar o problema:

- Se a opção do servidor IFILE foi especificada, verifique se o nome do nó é declarado no arquivo que ele especifica.
- Se o arquivo das interfaces existe no diretório `sqllib`, verifique se o nome do nó é declarado nele.
- Se a opção do servidor IFILE não estiver especificada e o arquivo de interfaces não existir em `sqllib`, verifique se o nome do nó é declarado no arquivo `%SYBASE%\ini\sql.ini` no Windows ou no arquivo `$SYBASE/interfaces` em UNIX.

Configurando o Acesso às Origens de Dados do Arquivo Estruturado em Tabela

Você pode integrar os dados que estão no arquivo estruturado em tabela com informações de outras origens utilizando um sistema federado.

Sobre Esta Tarefa

Para configurar um servidor federado para acessar origens de dados do arquivo estruturado em tabela, é necessário fornecer ao servidor federado informações sobre as origens de dados e objetos que você deseja acessar. Depois de configurar o servidor federado, você poderá criar consultas e utilizar as funções customizadas para acessar as origens de dados do arquivo estruturado em tabela.

Arquivos Estruturados em Tabela - Visão Geral

Um arquivo estruturado em tabela é um arquivo que possui uma estrutura regular que consiste em uma série de registros ou linhas de informações. Cada registros contém o mesmo número de campos. Os dados nos campos são separados por um delimitador, como uma vírgula.

O exemplo a seguir mostra o conteúdo de um arquivo chamado `DRUGDATA1.TXT`. Ele contém três registros, cada um com três campos, separados por vírgulas:

```
234,DrugnameA,Manufacturer1  
332,DrugnameB,Manufacturer2  
333,DrugnameC,Manufacturer2
```

O primeiro campo é o número do ID exclusivo para o medicamento. O segundo campo é o nome do medicamento. O terceiro campo é o nome do fabricante do medicamento.

O delimitador de campo pode ter mais de um caractere. As aspas simples não podem ser utilizadas como um delimitador. O delimitador deve ser consistente em todo o arquivo. Um valor nulo é representado por dois delimitadores próximos um do outro ou por um delimitador seguido por um terminador de linha, se o campo NULL for o último na linha. O delimitador de coluna não pode existir como um dado válido para uma coluna.

Por Exemplo:

```
234,DrugnameA,Manufacturer1  
332,DrugnameB,Manufacturer2  
333,DrugnameC,Manufacturer2  
356,,Manufacturer1
```

Atributos para Arquivos Estruturados em Tabela

Os registros ou linhas em arquivos estruturados em tabela podem ser classificados ou não classificados. O wrapper de arquivos estruturados em tabela podem procurar arquivos que estão classificados de maneira mais eficiente do que arquivos que não estão classificados.

Se os dados em um arquivo estruturado em tabela forem classificados, a classificação deverá estar em ordem crescente na coluna-chave. Você deve configurar a opção SORTED como Y na coluna-chave ao criar ou definir as colunas para o apelido. Caso contrário, o wrapper processará os dados no arquivo estruturado em tabela como dados não classificados.

Arquivos classificados

DRUGDATA1.TXT contém registros classificados. O arquivo é classificado pelo primeiro campo, o número do ID exclusivo para o medicamento. Este campo é a chave principal porque é exclusivo para cada medicamento. Os arquivos classificados devem ser classificados em ordem crescente.

```
234,DrugnameA,Manufacturer1  
332,DrugnameB,Manufacturer2  
333,DrugnameC,Manufacturer2
```

Arquivos não classificados

DRUGDATA2.TXT contém registros não classificados. Não existe nenhuma ordem na maneira que os registros são listados no arquivo.

```
556,DrugnameB,Manufacturer2  
234,DrugnameA,Manufacturer1  
721,DrugnameC,Manufacturer2
```

Wrapper de Arquivos Estruturados em Tabela

Os dados em um arquivo estruturado em tabela podem ser juntados com dados em outros arquivos estruturados em tabela, dados relacionais ou dados não relacionais e não estruturados.

Utilizando um wrapper, o servidor federado pode processar instruções SQL que consultam dados em um arquivo estruturado em tabela como se os dados estivessem contidos em uma tabela ou visualização relacional comum.

A maneira como o servidor federado funciona com arquivos estruturados em tabela é ilustrada na figura a seguir.

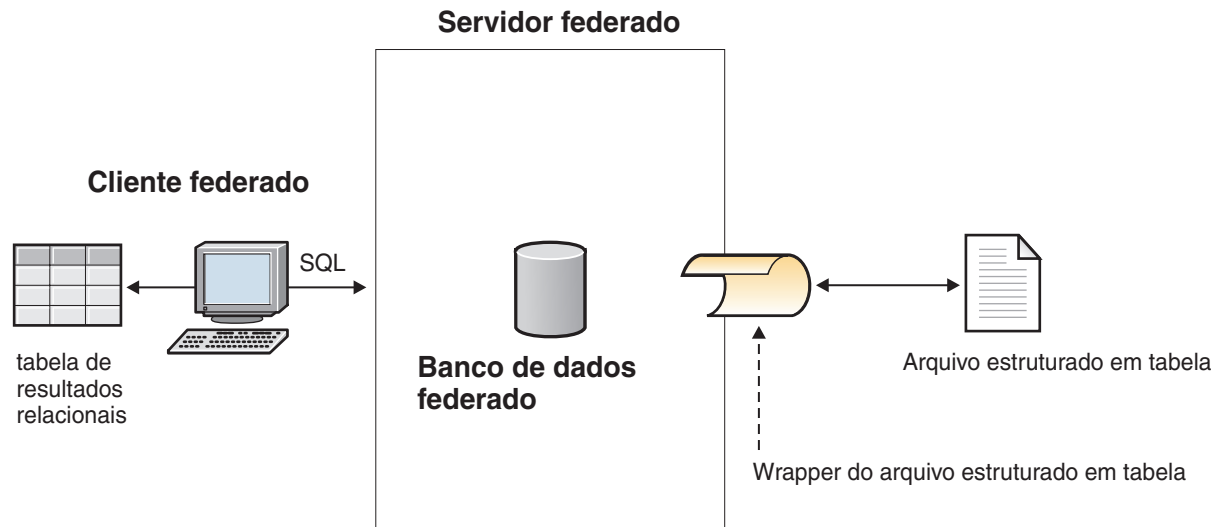


Figura 11. Como funciona o wrapper do arquivo estruturado em tabela

Por exemplo, o arquivo estruturado em tabela DRUGDATA2.TXT está localizado em um computador em seu laboratório. Os dados no arquivo são:

```
556,DrugnameB,Manufacturer1
234,DrugnameA,Manufacturer2
721,DrugnameC,Manufacturer2
```

É uma tarefa enfadonha tentar consultar e corresponder estes dados a outras tabelas a partir de outras origens de dados utilizadas.

Depois de registrar o arquivo DRUGDATA2.TXT no servidor federado, o wrapper do arquivo estruturado em tabela poderá acessar os dados no arquivo como se os dados estivessem em uma tabela relacional.

Por exemplo, você pode executar a seguinte consulta:

```
SELECT * FROM DRUGDATA2 ORDER BY DCODE
```

Esta consulta produz os seguintes resultados:

DCODE	DRUG	MANUFACTURER
234	DrugnameA	Manufacturer2
556	DrugnameB	Manufacturer1
721	DrugnameC	Manufacturer2

Você pode juntar os dados no arquivo DRUGDATA2.TXT com dados de outras origens de dados relacionais e não relacionais e analisar todos os dados juntos.

Incluindo Origens de Dados de Arquivo com Estrutura de Tabela em um Servidor Federado

Para configurar um servidor federado para acessar origens de dados do arquivo estruturado em tabela, é necessário fornecer ao servidor federado informações sobre as origens de dados e objetos que você deseja acessar.

Antes de Iniciar

- A federação deve ser instalada em um servidor que atuará como o servidor federado.
- Um banco de dados deve existir no servidor federado.

Procedimento

1. Registre o wrapper do arquivo estruturado em tabela.
2. Registre a definição do servidor do arquivo estruturado em tabela.
3. Registre os apelidos para os arquivos estruturados em tabela.

Registrando o Wrapper de Arquivo Estruturado em Tabela

Você deve registrar um wrapper para acessar as origens de dados de arquivos estruturados em tabela. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Procedimento

Emita a instrução CRIAR WRAPPER a partir da linha de comandos.

```
CREATE WRAPPER wrapper_name  
LIBRARY library_name;
```

Por exemplo, para registrar um wrapper com o nome `flat_files_wrapper` no servidor federado que utiliza o sistema operacional AIX, emita a seguinte instrução:

```
CREATE WRAPPER flat_files_wrapper  
LIBRARY 'libdb2lsfile.a';
```

Especifique o parâmetro LIBRARY na instrução CREATE WRAPPER. O nome do arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado. Consulte a lista de arquivos da instrução do wrapper do arquivo estruturado em tabela para obter o nome da biblioteca correto para especificar a instrução CREATE WRAPPER.

Arquivos da Biblioteca do Wrapper de Arquivos Estruturados em Tabela:

Os arquivos da biblioteca do wrapper de arquivos estruturados em tabelas são adicionados no servidor federado quando você instala o IBM InfoSphere Federation Server.

Ao instalar o IBM InfoSphere Federation Server, três arquivos de bibliotecas são incluídos no caminho do diretório padrão. Por exemplo, se o servidor federado estiver em execução no AIX, os arquivos da biblioteca do wrapper incluídos no caminho de diretório serão `libdb2lsfile.a`, `libdb2lsfileF.a` e `libdb2lsfileU.a`. O arquivo da biblioteca do wrapper padrão é `libdb2lsfile.a`. Os outros arquivos da biblioteca do wrapper são utilizados com opções do wrapper específicas.

Inclua o parâmetro LIBRARY na instrução CREATE WRAPPER e especifique o nome do arquivo da biblioteca padrão do wrapper.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 36. Locais e nomes de arquivos da biblioteca do cliente de arquivos estruturados em tabela

Sistema Operacional	Caminho do Diretório	Nomes de arquivos da biblioteca do wrapper
AIX	/usr/opt/<install_path>/lib32/ /usr/opt/<install_path>/lib64/	libdb2lsfile.a
Linux	/opt/IBM/db2/<install_path>/lib32 /opt/IBM/db2/<install_path>/lib64	libdb2lsfile.so
Solaris	/opt/IBM/db2/<install_path>/lib32 /opt/IBM/db2/<install_path>/lib64	libdb2lsfile.so
Windows	%DB2PATH%\bin	db2lsfile.dll

<install_path> é o caminho do diretório em que o IBM InfoSphere Federation Server está instalado no Linux ou UNIX.

%DB2PATH% é a variável de ambiente utilizada para especificar o caminho do diretório em que o IBM InfoSphere Federation Server está instalado no Windows. O caminho padrão do diretório do Windows é C:\Arquivos de Programa\IBM\SQLLIB.

Registrando a Definição de Servidor para Arquivos Estruturados em Tabela

Para arquivos estruturados em tabela, você deve registrar uma definição do servidor porque a hierarquia de objetos federados requer que os arquivos estruturados em tabela, que são identificados por apelidos, estejam associados a um objeto de definição do servidor específico.

Procedimento

Emita a instrução CRIAR SERVIDOR a partir da linha de comandos.

Por Exemplo:

```
CREATE SERVER server_definition_name
WRAPPER wrapper_name;
```

Instrução CREATE SERVER - Exemplo para o Wrapper do Arquivo Estruturado em Tabela:

Utilize a instrução CREATE SERVER para registrar a definição do servidor para o wrapper do arquivo estruturado em tabela. Este exemplo mostra os parâmetros necessários.

O exemplo a seguir mostra como registrar uma definição do servidor chamada *biochem_lab* para um arquivo de texto que contém dados biomédicos. A instrução CREATE SERVER que você deve emitir é:

```
CREATE SERVER biochem_lab WRAPPER flat_files_wrapper;
```

biochem_lab

O nome designado à definição do servidor do arquivo estruturado em tabela. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER *flat_files_wrapper*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

Registrando Apelidos para Arquivos Estruturados em Tabela

Você deve registrar um apelido para cada arquivo estruturado em tabela que deseja acessar. Utilize estes apelidos, em vez dos nomes dos arquivos, quando consultar origens de dados do arquivo estruturado em tabela.

Sobre Esta Tarefa

Restrições

- Se um campo não numérico for muito longo para seu tipo de coluna, os dados em excesso serão truncados.
- Se um campo decimal no arquivo tiver mais dígitos após o caractere raiz do que o permitido, os dados em excesso serão truncados. Por exemplo, se o número for 10.123456 e o tipo de dados especificar que são permitidos 3 dígitos após o caractere raiz, o número será truncado para 10.123. O caractere raiz é determinado pelo item RADIXCHAR da categoria de Suporte ao Idioma Nacional LC_NUMERIC. O parâmetro de escala para o tipo de dados da coluna especifica o número de dígitos permitidos após o caractere raiz.
- Se você tentar acessar as origens de dados do arquivo estruturado em tabela que estão em uma unidade compartilhada a partir do servidor federado que executa o Windows 2003, suas consultas poderão falhar. Esta é uma limitação do Windows 2003. Você pode evitar este problema especificando o caminho absoluto na opção FILE_PATH na instrução CREATE NICKNAME.
- O comprimento máximo de uma linha de dados é de 10 megabytes (10485760 bytes).
-

Ao criar um apelido para um arquivo estruturado em tabela, as informações nos dados do arquivo serão mapeadas para uma tabela relacional. Você cria apelidos para seu arquivo estruturado em tabela em uma de duas maneiras:

- Especificando o arquivo estruturado em tabela quando cria o apelido utilizando a opção de apelido FILE_PATH.
- Especificando o arquivo estruturado em tabela quando consulta a origem de dados, utilizando a opção de coluna do apelido DOCUMENT. Quando esta opção é utilizada, o apelido pode ser utilizado para representar dados de qualquer arquivo estruturado em tabela cujo esquema corresponde à definição de apelido.

Os nomes que você dá aos apelidos podem ter até 128 caracteres.

Procedimento

Emita a instrução CRIAR APELIDO a partir da linha de comandos.

Por Exemplo:

```
CREATE NICKNAME nickname
(
    column_name data_type,
    column_name data_type,
    column_name data_type
)
FOR SERVER server_definition_name
OPTIONS (nickname_options);
```

O que Fazer Depois

Repita esta etapa para cada arquivo estruturado em tabela para o qual deseja criar um apelido.

Instrução CREATE NICKNAME - Exemplos para o Wrapper do Arquivo Estruturado em Tabela:

Utilize a instrução CREATE NICKNAME para registrar um apelido para um arquivo estruturado em tabela que você deseja acessar.

Você deve especificar a opção de apelido FILE_PATH ou a opção de coluna de apelido DOCUMENT quando registrar um apelido para um arquivo estruturado em tabela.

Criando um Apelido com a Opção de Apelido FILE_PATH

O exemplo a seguir mostra uma instrução CREATE NICKNAME para o arquivo estruturado em tabela DRUGDATA1.TXT:

```
CREATE NICKNAME DRUGDATA1
(
  Dcode INTEGER NOT NULL,
  Drug CHAR(20),
  Manufacturer CHAR(20)
)
FOR SERVER biochem_lab
OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT')
```

DRUGDATA1

Um apelido exclusivo utilizado para identificar o arquivo estruturado em tabela.

O apelido é um nome de duas partes -o esquema e o apelido. Se você omitir o esquema quando registrar o apelido, o esquema do apelido será o ID de autorização do usuário que registra o apelido.

Dcode INTEGER NOT NULL

O nome e tipo de dados para uma coluna do arquivo estruturado em tabela que contém o código de um medicamento.

Drug CHAR(20)

O nome e tipo de dados para uma coluna do arquivo estruturado em tabela que contém o nome de um medicamento.

Manufacturer CHAR(20)

O nome e tipo de dados para uma coluna do arquivo estruturado em tabela que contém o nome de um fabricante de medicamento.

FOR SERVER *biochem_lab*

O nome designado à definição do servidor do arquivo estruturado em tabela na instrução CREATE SERVER.

FILE_PATH '/usr/pat/DRUGDATA1.TXT'

Especifica o caminho completo do diretório e nome do arquivo para o arquivo estruturado em tabela que contém os dados que você deseja acessar. O caminho deve ser colocado entre aspas simples.

Criando um Apelido com a Opção de Coluna de Apelido DOCUMENT

Ao criar um apelido utilizando a opção de coluna de apelido DOCUMENT, você está especificando que o nome do arquivo estruturado em tabela será fornecido

durante a execução de uma consulta que utiliza o apelido. Você pode especificar a opção de coluna de apelido DOCUMENT em apenas uma coluna quando registrar o apelido. A coluna associada à opção DOCUMENT deve ser um tipo de dados VARCHAR ou CHAR. É necessário incluir o caminho completo do arquivo ao executar uma consulta que utiliza o apelido.

O exemplo a seguir mostra uma instrução CREATE NICKNAME que especifica a opção de coluna de apelido DOCUMENT:

```
CREATE NICKNAME customers
(
  doc VARCHAR(100) OPTIONS(DOCUMENT 'FILE'),
  name VARCHAR(16),
  address VARCHAR(30),
  id VARCHAR(16)
FOR SERVER biochem_lab
```

customers

Um nome exclusivo para o apelido.

O apelido é um nome de duas partes -o esquema e o apelido. Se você omitir o esquema quando registrar o apelido, o esquema do apelido será o ID de autorização do usuário que registra o apelido.

doc VARCHAR(100) OPTIONS(DOCUMENT 'FILE')

O nome e tipo de dados para uma coluna utilizada para especificar o nome do arquivo estruturado em tabela que você deseja acessar. Você especifica o nome do arquivo quando executa a consulta.

name VARCHAR(16)

O nome e tipo de dados para uma coluna do arquivo estruturado em tabela que contém o nome do cliente.

address VARCHAR(30)

O nome e tipo de dados para uma coluna do arquivo estruturado em tabela que contém o endereço do cliente.

id VARCHAR(16)

FOR SERVER *biochem_lab*

O nome designado à definição do servidor do arquivo estruturado em tabela na instrução CREATE SERVER.

FILE_PATH *'/usr/pat/DRUGDATA1.TXT'*

Especifica o caminho completo do diretório e nome do arquivo para o arquivo estruturado em tabela que contém os dados que você deseja acessar. Você deve especificar a opção de apelido FILE_PATH ou DOCUMENT na instrução CREATE NICKNAME. O caminho deve ser colocado entre aspas simples.

Criando um Apelido com os Parâmetros Opcionais

O exemplo a seguir mostra uma instrução CREATE NICKNAME para o arquivo estruturado em tabela DRUGDATA1.TXT:

```
CREATE NICKNAME DRUGDATA1
(
  Dcode INTEGER NOT NULL,
  Drug CHAR(20),
  Manufacturer CHAR(20)
)
FOR SERVER biochem_lab
OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT',
```

```
COLUMN_DELIMITER ',' ,'  
SORTED 'Y',  
KEY_COLUMN 'DCODE',  
VALIDATE_DATA_FILE 'Y')
```

DRUGDATA1

Um apelido exclusivo utilizado para identificar o arquivo estruturado em tabela.

O apelido é um nome de duas partes -o esquema e o apelido. Se você omitir o esquema quando registrar o apelido, o esquema do apelido será o ID de autorização do usuário que registra o apelido.

Dcode *INTEGER NOT NULL*

O nome e tipo de dados para uma coluna do arquivo estruturado em tabela que contém o código de um medicamento.

Drug *CHAR(20)*

O nome e tipo de dados para uma coluna do arquivo estruturado em tabela que contém o nome de um medicamento.

Manufacturer *CHAR(20)*

O nome e tipo de dados para uma coluna do arquivo estruturado em tabela que contém o nome de um fabricante de medicamento.

FOR SERVER *biochem_lab*

O nome designado à definição do servidor do arquivo estruturado em tabela na instrução CREATE SERVER.

FILE_PATH *'/usr/pat/DRUGDATA1.TXT'*

Especifica o caminho completo do diretório e nome do arquivo para o arquivo estruturado em tabela que contém os dados que você deseja acessar. Você deve especificar a opção de apelido FILE_PATH ou DOCUMENT na instrução CREATE NICKNAME. O caminho deve ser colocado entre aspas simples.

COLUMN_DELIMITER *','*

Especifica o delimitador utilizado para separar os campos em um arquivo estruturado em tabela. O valor do delimitador deve ser colocado entre aspas simples. O delimitador de coluna pode ter mais de um caractere de comprimento. Se você não especificar um delimitador de coluna, o delimitador padrão será uma vírgula. Uma aspa simples não pode ser utilizada como um delimitador. O delimitador de coluna deve ser consistente em todo o arquivo. Um valor nulo é representado por dois delimitadores próximos um do outro ou por um delimitador seguido por um terminador de linha, se o campo NULL for o último na linha. O delimitador de coluna não pode existir como um dado válido para uma coluna. Por exemplo, um delimitador de coluna de uma vírgula não poderá ser utilizado se uma das colunas contiver dados com vírgulas incorporadas.

SORTED *'Y'*

Especifica que o arquivo de origem de dados é classificado. As origens de dados classificadas devem ser classificadas em ordem crescente de acordo com a seqüência de intercalação para o código do idioma atual, conforme definido pelas configurações na categoria Suporte ao Idioma Nacional LC_COLLATE. Se você especificar que a origem de dados é classificada, configure a opção VALIDATE_DATA_FILE como 'Y'. O valor padrão para o parâmetro SORTED é 'N'.

KEY_COLUMN *'DCODE'*

O nome da coluna o arquivo que forma a chave na qual o arquivo está

classificado. O valor da coluna-chave deve ser colocado entre aspas simples. Especifique esta opção apenas se especificar a opção de apelido SORTED. Uma coluna que é designada com a opção de coluna de apelido DOCUMENT não deve ser especificada como a coluna-chave. O valor deve ser o nome de uma coluna que é definido na instrução CREATE NICKNAME. A coluna-chave não deve conter valores nulos. Apenas chaves de colunas simples são suportadas. Não são permitidas várias chaves de colunas. A coluna deve ser classificada em ordem crescente. Se o valor não for especificado para um apelido classificado, será assumido como padrão a primeira coluna no arquivo de apelido.

VALIDATE_DATA_FILE 'Y'

Especifica se o wrapper verifica se a coluna-chave está classificada em ordem crescente e verifica chaves nulas. Os valores válidos para esta opção são Y ou N. Esta opção não será permitida se a opção da coluna de apelido for utilizada para o caminho de arquivo.

Designando Colunas-chave ao Registrar um Apelido

Você pode designar uma coluna-chave especificando a restrição NOT NULL na instrução de apelido:

```
CREATE NICKNAME tox (tox_id INTEGER NOT NULL, toxicity VARCHAR(100))
FOR SERVER tox_server1
  OPTIONS (FILE_PATH'/tox_data.txt', SORTED 'Y')

CREATE NICKNAME weights (mol_id INTEGER, wt VARCHAR(100) NOT NULL)
FOR SERVER wt_server
  OPTIONS (FILE_PATH'/wt_data.txt', SORTED 'Y', KEY_COLUMN 'WT')
```

NOT NULL

Especifica que a coluna não pode conter valores nulos ou em branco.

O wrapper não aplica a restrição NOT NULL, mas o banco de dados federado sim. Se você criar um apelido e anexar uma restrição NOT NULL em uma coluna e, em seguida, selecionar uma linha contendo um valor nulo para a coluna, o banco de dados federado emitirá um erro SQL0407N indicando que não é possível designar um valor NULL a uma coluna NOT NULL.

A exceção a esta regra é para apelidos classificados. A coluna-chave para apelidos classificados não pode ser NULL. Se for localizada uma coluna-chave NULL para um apelido classificado, será emitido o erro SQL1822N, indicando que a coluna-chave está ausente.

Nomes de Colunas que Fazem Distinção entre Maiúsculas e Minúsculas

O banco de dados federado altera os nomes de colunas para maiúsculas, a menos que estejam definidos com aspas duplas. O exemplo a seguir não funcionará corretamente, porque o valor da opção KEY_COLUMN está colocado entre aspas simples. No exemplo, o nome da coluna será convertido pelo banco de dados federado em EMPNO. Portanto, quando você especificar empno em uma consulta, a coluna não será reconhecida pelo banco de dados federado.

```
CREATE NICKNAME depart (
  empno char(6) NOT NULL)
FOR SERVER DATASTORE
  OPTIONS(FILE_PATH'data.txt', SORTED 'Y', KEY_COLUMN 'empno');
```


Servidores Federados do Windows 2003

Se você tentar acessar origens de dados do arquivo estruturado em tabela que estão em uma unidade compartilhada a partir de um servidor federado que executa o Windows 2003, sua consulta poderá falhar com a seguinte mensagem de erro:

```
SQL1822N Código de erro inesperado "ERRNO = 2" recebido da origem de dados
"SERVERNAME1". O texto e tokens associados são "Não foi possível ler o arquivo".
SQLSTATE=560BD
```

Esta é uma limitação do Windows 2003. Você pode evitar este problema especificando o caminho absoluto na opção FILE_PATH na instrução CREATE NICKNAME.

O exemplo a seguir mostra uma instrução CREATE NICKNAME com um caminho abreviado especificado na opção FILE_PATH:

```
CREATE NICKNAME nickname
(
  COL1 CHAR (10) NOT NULL
)
FOR SERVER servername1
OPTIONS (FILE_PATH 'X:\textfile1.txt');
```

em que X:\ é a unidade mapeada para a máquina remota. As consultas que utilizam este apelido podem falhar porque você especificou o caminho abreviado.

Para o servidor federado que executa o Windows 2003, especifique o caminho absoluto na opção FILE_PATH na instrução CREATE NICKNAME.

Por exemplo:

```
CREATE NICKNAME nickname
(
  COL1 CHAR (10) NOT NULL
)
FOR SERVER servername1
OPTIONS (FILE_PATH '\\host.svl.ibm.com\D$\textfile1.txt');
```

Modelo de Controle de Acesso de Arquivo para o Wrapper de Arquivo Estruturado em Tabela

O wrapper acessa arquivos estruturados em tabela utilizando as informações de autorização do proprietário da instância de banco de dados federado. O wrapper pode acessar apenas arquivos que podem ser lidos por este ID do usuário ou ID do grupo. O ID de autorização que estabelece a conexão com o banco de dados federado não é utilizado para acessar arquivos estruturados em tabela.

Em um servidor federado, qualquer arquivo estruturado em tabela para o qual foi criado um apelido deve estar acessível com o mesmo nome do caminho a partir de cada nó. O arquivo não precisa estar em um nó de banco de dados federado, desde que o arquivo possa ser acessado a partir de qualquer nó com um caminho comum.

Para acessar um arquivo estruturado em tabela, o wrapper precisa de uma identidade do usuário para fins de segurança. O wrapper do arquivo estruturado em tabela utiliza a identidade do usuário que está associada ao serviço do banco de dados federado. O nome do serviço do banco de dados federado depende do nome da instância do banco de dados. Por exemplo, se o nome da instância de

banco de dados for DB2, o nome do serviço será DB2 - DB2. Para determinar a identidade do usuário que está associada a um serviço do banco de dados federado, utilize o Painel de Controle no Windows para exibir os serviços. Dê um clique duplo no nome do serviço para exibir a página de propriedades de Logon.

Arquivos Estruturados em Tabela Localizados em Unidades Remotas

Os arquivos estruturados em tabela que você deseja acessar devem estar em uma unidade local ou mapeada.

Redes que possuem um domínio do Windows configurado

A conta de logon para o serviço de banco de dados federado deve ser uma conta do domínio que possui acesso à pasta compartilhada na unidade mapeada na qual os arquivos estruturados em tabela residem.

Redes que não possuem um domínio do Windows configurado

A conta de logon do serviço de banco de dados federado deve ter o mesmo nome de usuário e senha que um usuário válido no computador que compartilha essa pasta. Esse usuário deve estar na lista de permissões para a pasta compartilhada com pelo menos acesso de leitura

Orientações para Otimizar o Desempenho de Consultas para o Wrapper do Arquivo Estruturado em Tabela

Você pode aprimorar o desempenho de consultas do arquivo estruturado em tabela tendo os arquivos que estão classificados e criando estatísticas para seus apelidos.

Utilize as seguintes dicas para aprimorar o desempenho de consultas:

- Classifique os dados em seus arquivos. O servidor federado pode procurar arquivos de maneira muito mais eficiente do que arquivos que não estão classificados.
- Para arquivos classificados, especifique um valor ou intervalo para a coluna-chave quando enviar uma consulta.
- As estatísticas para apelidos de arquivos estruturados em tabela devem ser atualizadas manualmente atualizando as visualizações SYSSTAT e SYSCAT. Utilize o recurso de atualização de estatísticas de Apelidos para atualizar as estatísticas para os apelidos do arquivo estruturado em tabela.

Configurando o Acesso às Origens de Dados Teradata

Para configurar um servidor federado para acessar origens de dados Teradata, você deve fornecer ao servidor federado informações sobre as origens de dados e os objetos que deseja acessar.

Antes de Iniciar

- O software do cliente Teradata deve ser instalado e configurado no servidor que atua como o servidor federado.
- A federação deve ser instalada em um servidor que atue como o servidor federado.
- Verifique a configuração do servidor federado.
- Verifique o parâmetro federado para assegurar que a federação está ativada.

Procedimento

1. Teste a conexão com o servidor Teradata.

2. Verifique se a biblioteca Teradata está ativada para link de tempo de execução (AIX).
3. Configure as variáveis de ambiente para o wrapper Teradata.
4. Registre o wrapper.
5. Registre a definição de servidor.
6. Crie os mapeamentos de usuário.
7. Teste a conexão com o servidor Teradata.
8. Registre apelidos para tabelas e visualizações do Teradata.

Testando a Conexão com o Servidor Teradata

Teste a conexão com o servidor Teradata para verificar se o software do cliente Teradata está corretamente configurado no servidor federado.

Antes de Iniciar

O utilitário BTEQ (Basic Teradata Query) e o Teradata Data Connector Application Program Interface (PIOM) devem ser instalados no servidor federado. O utilitário BTEQ e o Teradata Data Connector Application Program Interface são instalados no servidor federado quando você instala o software do cliente Teradata.

Sobre Esta Tarefa

Você utiliza o utilitário BTEQ para enviar uma consulta SQL para verificar se o servidor federado pode se conectar ao servidor Teradata. Consulte a documentação do Teradata para obter informações adicionais sobre o utilitário BTEQ.

Procedimento

1. Inicie a sessão do utilitário BTEQ e efetue logon no servidor Teradata.
2. Emita um comando SQL para verificar se é possível conectar-se com êxito ao servidor Teradata.

Por exemplo:

```
select count(*) from dbc.tables;
```

Se a conexão for bem-sucedida, você visualizará a saída da consulta.

Se a conexão não for bem-sucedida, você receberá um erro. Verifique o software do cliente Teradata para verificar se ele está corretamente instalado e configurado no servidor federado.

3. Efetue logoff do servidor Teradata e encerre a sessão do utilitário BTEQ.

O que Fazer Depois

Após concluir esta tarefa, verifique se a biblioteca do Teradata está ativada para link de tempo de execução.

Verificando se a Biblioteca do Teradata Está Ativada para Link de Tempo de Execução (AIX)

Quando você incluir uma origem de dados Teradata em seu servidor federado no AIX, deve verificar se o link de tempo de execução está ativado antes de registrar wrappers ou servidores.

Procedimento

1. Vá para o diretório no qual reside o arquivo libcliv2.so.

O arquivo `libcliv2.so` está instalado com o software do cliente Teradata. Por padrão, ele está instalado no diretório `/usr/lib`.

2. Em um prompt de comandos, emita o seguinte comando UNIX para verificar se o link de tempo de execução está ativado:

```
dump -H libcliv2.so | grep libtli.a
```
3. Verifique os nomes de arquivo que são retornados. Se o arquivo `libtli.a` é retornado, a biblioteca Teradata está ativada para link de tempo de execução. Se o nome do arquivo `libtli.a` não é retornado, abra uma janela de comandos e emita os seguintes comandos UNIX para ativar o link de tempo de execução para a biblioteca Teradata:

```
rtl_enable libcliv2.so -F libtli.a
mv libcliv2.so libcliv2.so.old
mv libcliv2.so.new libcliv2.so
chmod a+r libcliv2.so
```

O que Fazer Depois

Depois de concluir essa tarefa, é possível configurar as variáveis de ambiente.

Definindo as Variáveis de Ambiente do Teradata

As variáveis de ambiente Teradata devem ser configuradas no arquivo `db2dj.ini` no servidor federado.

Sobre Esta Tarefa

Restrições

Revise as restrições ao arquivo `db2dj.ini`.

O arquivo `db2dj.ini` contém informações de configuração sobre o software do cliente Teradata que está instalado no seu servidor federado.

Há variáveis de ambiente requeridas e opcionais para as origens de dados Teradata.

Se você instalou o software do cliente Teradata antes de ter instalado o wrapper Teradata, as variáveis de ambiente Teradata requeridas estão configuradas no arquivo `db2dj.ini`.

Você deve configurar as variáveis de ambiente utilizando as etapas nesta tarefa se não instalou o software do cliente Teradata antes de ter instalado o wrapper Teradata ou se desejar configurar qualquer uma das variáveis de ambiente opcionais.

Procedimento

1. Utilize um dos seguintes métodos:

Método	Etapas
Configurar as variáveis de ambiente automaticamente.	Execute o assistente de instalação do IBM InfoSphere Federation Server. Siga as instruções no assistente. Importante: Configure as variáveis de ambiente necessárias executando o assistente de instalação. As variáveis de ambiente opcionais devem ser definidas manualmente.

Método	Etapa
Configure as variáveis de ambiente com a ferramenta de configuração do wrapper.	Execute a ferramenta de configuração do wrapper para configurar as variáveis de ambiente necessárias e as opcionais. Follow the instructions in "Executando a Ferramenta de Configuração de Wrapper" na página 1.
Configurar manualmente as variáveis de ambiente.	<p>Edite o arquivo db2dj.ini:</p> <p>O arquivo db2dj.ini está localizado no diretório que a variável de registro DB2_DJ_INI do DB2 especifica. Quando a variável DB2_DJ_INI não está configurada, o arquivo db2dj.ini corresponde a um dos seguintes caminhos padrão, dependendo do sistema operacional:</p> <ul style="list-style-type: none"> No UNIX: <i>instancehome</i>/sql1lib/cfg/db2dj.ini. <i>instancehome</i> O diretório inicial do proprietário da instância. Em Windows: %DB2PATH%\cfg\db2dj.ini %DB2PATH% O diretório em que o sistema de banco de dados do DB2 está instalado, por exemplo, C:\Program Files\IBM\sql1lib. <p>Se o arquivo não existir, você poderá criar um arquivo com o nome db2dj.ini utilizando qualquer editor de texto. No arquivo db2dj.ini, você deve especificar o caminho completo no valor das variáveis de ambiente; caso contrário, serão encontrados erros. Por Exemplo: COPLIB=/usr/lib</p>

- Configure as variáveis de ambiente de conversão de página de códigos do Teradata (conforme necessário).
- Para assegurar que as variáveis de ambiente estejam configuradas no servidor federado, recicle a instância do banco de dados federado com esses comandos:
db2stop
db2start

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar o wrapper.

Variáveis de Ambiente Teradata

Há variáveis de ambiente requeridas e opcionais para as origens de dados Teradata. Essas variáveis são definidas no arquivo db2dj.ini.

As seguintes variáveis de ambiente são válidas para o Teradata:

- COPLIB
- COPERR
- TERADATA_CHARSET (opcional)
- NETRACE (opcional)
- COPANOMLOG (opcional)

Descrições da Variável

COPLIB

Specifies the directory path on the federated server for the CLI system

parameter block file `clispb.dat`. You must specify the fully qualified path for the `COPLIB` variable. The Teradata CLI installation directory of the `clispb.dat` file is `/usr/lib`, if you modify the directory path of the file, you must ensure that you update this variable:

```
COPLIB=/usr/lib
```

For 64-bit clients, if you must run `djxlinkTeradata.sh` and the library is not located in the default location, then you must set the `TERADATA_LIB_DIR` environment variable to the directory location of the 64-bit client library. Depending on your operating system, the default locations for the library are:

- AIX: `/usr/lib/lib_64/libcliv2.so`
- HP-UX:
 - PA-RISC 64-bit: `/usr/lib/pa20_64/libcliv2.sl`
 - Itanium Processor Family (IPF) 64-bit: `/usr/lib/hpux64/libcliv2.so`
- Linux: `/usr/lib64/libcliv2.so`
- Solaris-SPARC: `/usr/lib/sparcv9/libcliv2.so`

Both the `clispb.dat` and error message catalog `errmsg.cat` files typically reside in the same directory.

COPERR

Especifica o caminho do diretório no servidor federado para o arquivo `errmsg.cat`. Especifique o caminho completo para a variável `COPERR`.

Por Exemplo:

```
COPERR=/usr/lib
```

TERADATA_CHARSET

Especifica o conjunto de caracteres de página de códigos a ser utilizado com as origens de dados Teradata.

A cada vez que o servidor federado se conecta com uma origem de dados Teradata, o wrapper Teradata determina qual conjunto de caracteres de página de código utilizar para a conexão. O wrapper Teradata pode configurar o conjunto de caracteres de página de códigos ou você pode designar uma página de códigos configurando a variável de ambiente `TERADATA_CHARSET`.

Se a variável de ambiente `TERADATA_CHARSET` estiver configurada no arquivo `db2dj.ini` no servidor federado, o wrapper utiliza o conjunto de caracteres da página de códigos no arquivo `db2dj.ini`. O valor na variável de ambiente `TERADATA_CHARSET` não está validado, mas se a variável de ambiente não estiver configurada com um valor válido, a origem de dados Teradata retorna um erro.

Se a variável de ambiente `TERADATA_CHARSET` não estiver configurada no arquivo `db2dj.ini` no servidor federado, o wrapper detecta o conjunto de caracteres do cliente com base na página de códigos do banco de dados.

Em servidores federados que executam UNIX, os seguintes valores são válidos para a variável de ambiente `TERADATA_CHARSET`:

- `HANGULKSC5601_2R4`
- `KanjiEUC_0U`
- `LATIN1_0A`
- `LATIN9_0A`
- `LATIN1252_0A`

- SCHGB2312_1T0
- TCHBIG5_1R0
- UTF8
- ASCII

Em servidores federados que executam Windows, os seguintes valores são válidos para a variável de ambiente TERADATA_CHARSET:

- HANGULKSC5601_2R4
- KanjiSJIS_0S
- LATIN1_0A
- LATIN1252_0A
- SCHGB2312_1T0
- TCHBIG5_1R0
- UTF8
- ASCII

NETRACE

Opcional. Ativa o recurso de rastreamento do software cliente Teradata. Esta variável é necessária apenas para depuração.

COPANOMLOG

Opcional. Ativa o recursos de criação de log do software cliente Teradata. Esta variável é necessária apenas para depuração.

Verificando o Conjunto de Caracteres no Servidor Teradata

Se o conjunto de caracteres correto não estiver especificado no servidor Teradata, você pode receber erros de conexão. Verifique se o conjunto de caracteres que você deseja utilizar está instalado no servidor Teradata.

Sobre Esta Tarefa

Procedimento

Para verificar se o conjunto de caracteres que você deseja utilizar está instalado no servidor Teradata:

Procedimento

1. Efetue logon no servidor Teradata utilizando o utilitário BTEQ ou qualquer outro utilitário de logon válido.
2. Emita a instrução a seguir para exibir a tabela dbc.chartranslations: `select * from dbc.chartranslations;`
3. Verifique o valor na terceira coluna, InstallFlag, da tabela que é retornada. O valor 'Y' na terceira coluna indica que o conjunto de caracteres está instalado e sendo utilizado no servidor Teradata.

Utilize a seguinte tabela para determinar se você possui o conjunto de caracteres correto instalado:

Tabela 37. Conjuntos de Caracteres para o Teradata

Conjunto de caracteres de duplo byte	Conjunto de caracteres de byte único	Conjunto de Caracteres Teradata	Idioma	Conjunto de Códigos do IBM DB2
941	897	"KanjiSJIS_0S"	Japonês	IBM-943
1362	1126	"HANGULKSC5601_2R4"	Coreano	1363

Tabela 37. Conjuntos de Caracteres para o Teradata (continuação)

Conjunto de caracteres de duplo byte	Conjunto de caracteres de byte único	Conjunto de Caracteres Teradata	Idioma	Conjunto de Códigos do IBM DB2
1385	1114	"SCHGB2312_1T0"	Chinês Simplificado	GBk
380	1115	"SCHGB2312_1T0"	Chinês Simplificado	IBM-1381
947	1114	"TCHBIG5_1R0"	Chinês Tradicional	big5
1200	1208	"UTF8"	Unicode	UTF-8
0	819	"Latin1_0A"	Inglês (Latin 1)	ISO8859-1
0	1252	"Latin1252_0A"	Inglês (Win Latin)	ISO8859-1/15
0	819	"ASCII"	Inglês (ASCII)	ISO8859-1

4. Se você não possui o conjunto de caracteres requerido instalado, instale o conjunto de caracteres para utilizar o wrapper do Teradata.
 - ASCII está ativado no servidor Teradata, mas ele não está catalogado na tabela `dbc.chartranslations`. Se todos os valores que são retornados para `InstallFlag` forem 'N', ASCII é o único conjunto de caracteres válidos no servidor Teradata e a variável de ambiente `TERADATA_CHARSET` deve ser configurada como ASCII no arquivo `db2dj.ini`.
 - Se o conjunto de caracteres que você deseja utilizar está listado na tabela `dbc.chartranslations`, mas o valor `InstallFlag` está configurado como 'N', emita a seguinte instrução para alterar `InstallFlag` como 'Y':


```
update dbc.chartranslations
       set installflag='Y' where CharSetName= 'character_set_name';
```
 - Se o conjunto de caracteres que você deseja utilizar não está listado na tabela `dbc.chartranslations`, entre em contato com o atendimento ao cliente da Teradata.
5. Reinicie o servidor Teradata para atualizar a lista de conjuntos de caracteres. Em uma janela de comandos Teradata, digite: **tpareset -f reason_for_restart**

Resolução de Problemas de Conjuntos de Caracteres para Origens de Dados Teradata:

Ao configurar a variável de ambiente `TERADATA_CHARSET` para uma origem de dados Teradata, você pode encontrar erros se o conjunto de caracteres correto não estiver especificado.

Sintoma

Se o conjunto de caracteres correto não estiver especificado para a origem de dados Teradata, você encontrará o seguinte erro:

```
SQL 1822N Código de erro inesperado "227"
recebido da origem de dados "<string>". O texto e os
tokens associados são "MTDP: EM_CHARNAME(227): nome de conjunto de caracteres
inválidos especificado". SQLSTATE=560BD
```

Causa

O conjunto de caracteres especificado na variável de ambiente `TERADATA_CHARSET` está incorreto.

Solução do Problema

Verifique se o conjunto de caracteres correto está instalado e especificado no servidor Teradata e no arquivo db2dj.ini.

Registrando o Wrapper Teradata

Você deve registrar um wrapper para acessar origens de dados Teradata. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Procedimento

Issue the CREATE WRAPPER statement from the command line and specify the default name for the Teradata wrapper.

Por exemplo:

```
CREATE WRAPPER TERADATA;
```

Lembre-se: Ao registrar o wrapper utilizando o nome padrão, TERADATA, o servidor federado utiliza automaticamente a biblioteca de wrapper Teradata apropriada para o sistema operacional no qual seu servidor federado está em execução.

Se o nome padrão do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, você poderá substituir o nome padrão do wrapper por um de sua escolha. Se você não utilizar o nome padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.

Por exemplo, para registrar um wrapper com o nome tera_wrapper em um servidor federado que utiliza o AIX, emita a seguinte instrução:

```
CREATE WRAPPER tera_wrapper  
  LIBRARY 'libdb2teradata.a'  
  OPTIONS (DB2_FENCED 'Y');
```

O arquivo da biblioteca do wrapper que você especifica depende do sistema operacional do servidor federado.

Opções obrigatórias para AIX e Solaris

Nota: A opção de wrapper DB2_FENCED na instrução CREATE WRAPPER é obrigatória para AIX e Solaris porque o wrapper Teradata suporta apenas clientes de 32 bits.

Por Exemplo:

```
CREATE WRAPPER TERADATA  
  OPTIONS (DB2_FENCED 'Y')
```

O que Fazer Depois

Depois de concluir essa tarefa, é possível registrar a definição do servidor.

Arquivos de Biblioteca do Wrapper Teradata

Os arquivos de biblioteca do wrapper Teradata são incluídos no servidor federado quando você instala o wrapper.

Ao instalar o wrapper Teradata, três arquivos de biblioteca são incluídos no caminho do diretório padrão. Por exemplo, se o servidor federado está em execução no AIX, os arquivos de biblioteca do wrapper que estão incluídos no caminho do diretório são libdb2teradata.a, libdb2teradataF.a e

libdb2teradataU.a. O arquivo de biblioteca do wrapper padrão é libdb2teradata.a. Os outros arquivos de biblioteca do wrapper são utilizados internamente pelo wrapper Teradata.

Se você não utilizar o nome do wrapper padrão ao registrá-lo, deve incluir o parâmetro LIBRARY na instrução CREATE WRAPPER e especificar o nome do arquivo de biblioteca do wrapper padrão.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 38. Locais de Biblioteca e Nomes de Arquivo do Wrapper Teradata

Sistema Operacional	Caminho do Diretório	Nomes do Arquivo de Biblioteca
AIX	/usr/opt/install_path/lib32/ /usr/opt/install_path/lib64/	libdb2teradata.a
Solaris	/opt/IBM/db2/install_path/lib32 /opt/IBM/db2/install_path/lib64	libdb2teradata.so
Windows	%DB2PATH%\bin	db2teradata.dll

install_path é o caminho diretório no qual o servidor federado está instalado no UNIX.

Registrando as Definições de Servidor para uma Origem de Dados Teradata

Você deve registrar cada servidor Teradata que deseja acessar no banco de dados federado.

Procedimento

1. Localize o arquivo hosts.
 - Nos servidores federados que executam AIX, o arquivo hosts está localizado no diretório /etc/hosts.
 - Em servidores federados que executam Windows, o arquivo hosts está localizado no diretório %WINDIR%\system32\drivers\etc\hosts .
2. Locate the server name of your Teradata node entry in the hosts file.

The server name is listed along with a suffix in the hosts file: *dbcnameCOPn*.

You specify the server name *dbcname* as the *node_name* variable of the CREATE SERVER statement. Use the following syntax of a Teradata node entry to locate the server name *dbcname*:

Sintaxe

```
a.b.c.d host_name dbcnameCOPn aliasnameCOPn
```

Parâmetros

a.b.c.d

The IP address of Teradata Server.

host_name

Opcional. The host name of Teradata server.

dbcnameCOPn

The server name and number of the application processor in a Teradata Communication Processor (COP) group. In systems that includes multiple Teradata servers and a single database instance, the servers are referred to as a COP group. Each server in a COP

group is assigned a unique sequential number, starting at 1. For example, if only one Teradata server exists, the server name and number of the application processor is *dbcnameCOP1*.

dbcname

The name of your Teradata server. The name is not longer than eight characters. For servers in a COP group, the servers have the same *dbcname* and sequential *COPn* values.

n The number of the application processor.

alias_nameCOPn

Opcional. The alternate name for *dbcname* and number of the application processor in a COP group. The number of the application processor is the same number that exists for the server name *dbcnameCOPn*.

Exemplo

In this sample hosts file, nodexyz is the server name and must be specified as the *node_name* variable of the CREATE SERVER statement:

```
127.0.0.1      localhost

9.22.5.77     nodexyz      nodexyzCOP1    # teradata server

9.66.111.133  rtplib05.data.xxx.com aap
9.66.111.161  rtpscm11.data.xxx.com aaprwr
9.66.111.161  rtpscm11.data.xxx.com accessm
```

3. To create the server, ssue the CREATE SERVER statement from the command line.

Por Exemplo:

```
CREATE SERVER server_definition_name
  TYPE TERADATA
  VERSION version_number
  WRAPPER wrapper_name
  OPTIONS (NODE 'node_name');
```

Although the *node_name* variable is specified as an option in the CREATE SERVER statement, this option is required for Teradata data sources.

Instrução CREATE SERVER - Exemplos para o Wrapper Teradata

Utilize a instrução CREATE SERVER para registrar definições de servidor para o wrapper Teradata. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo com opções adicionais do servidor.

O exemplo a seguir mostra como registrar uma definição de servidor para um wrapper Teradata emitindo a instrução CREATE SERVER:

```
CREATE SERVER tera_server TYPE TERADATA
  VERSION 2.5 WRAPPER my_wrapper
  OPTIONS (NODE 'tera_node');
```

tera_server

Um nome que você designa para o servidor de banco de dados Teradata. Nomes de definição de servidor duplicados não são permitidos.

TYPE TERADATA

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso. Para o wrapper Teradata, o tipo de servidor deve ser TERADATA.

VERSION 2.5

A versão do servidor de banco de dados Teradata que você deseja acessar.

WRAPPER TERADATA

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

NODE 'tera_node'

O nome do nó no qual o servidor de banco de dados Teradata reside. Obtenha o nome do nó no arquivo hosts. Este valor faz distinção entre maiúsculas e minúsculas.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para as origens de dados Teradata.

Opções do Servidor

Quando você cria uma definição do servidor, é possível especificar opções adicionais do servidor na instrução CREATE SERVER. As opções do servidor podem ser opções do servidor gerais e opções do servidor específicas do Teradata.

As opções do servidor CPU_RATIO e IO_RATIO fornecem as informações estatísticas sobre o servidor Teradata para o otimizador de consulta. Para especificar que os recursos de CPU do servidor federado sejam duas vezes mais eficientes que os recursos de CPU do servidor Teradata, especifique o valor da opção de servidor CPU_RATIO como 2.0. Para especificar que os dispositivos de E/S do servidor federado processem dados três vezes mais rápido do que os dispositivos de E/S do servidor Teradata, configure a opção de servidor IO_RATIO como 3.0.

O seguinte exemplo mostra uma definição de servidor Teradata com estas opções:

```
CREATE SERVER tera_server TYPE TERADATA
  VERSION 2.5 WRAPPER my_wrapper
  OPTIONS (NODE 'tera_node', CPU_RATIO '2.0', IO_RATIO '3.0');
```

Criando o Mapeamento de Usuário para uma Origem de Dados Teradata

Ao tentar acessar um servidor Teradata, o servidor federado estabelece uma conexão com o servidor Teradata utilizando um ID de usuário e senha que são válidos para a origem de dados.

Sobre Esta Tarefa

Crie um mapeamento de usuário para cada ID do usuário que acessará o sistema federado para enviar pedidos distribuídos para a origem de dados Teradata.

Procedimento

Emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR local_userID SERVER server_definition_name
  OPTIONS (REMOTE_AUTHID 'remote_userID', REMOTE_PASSWORD 'remote_password')
```

Embora as variáveis REMOTE_AUTHID e REMOTE_PASSWORD estejam especificadas como opções na instrução CREATE USER MAPPING, estas opções são necessárias para acessar as origens de dados Teradata.

O que Fazer Depois

Após concluir esta tarefa, teste a conexão do servidor federado com o servidor Teradata.

Instrução CREATE USER MAPPING - Exemplos para o Wrapper Teradata

Utilize a instrução CREATE USER MAPPING para mapear um ID de autorização do servidor federado para um ID do usuário e senha remotos do Teradata. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o USER de registro especial do DB2 com a instrução CREATE USER MAPPING.

O exemplo a seguir mostra como mapear um ID de autorização local para um ID do usuário e senha remotos do Teradata:

```
CREATE USER MAPPING FOR MICHAEL SERVER tera_server
  OPTIONS (REMOTE_AUTHID 'mike', REMOTE_PASSWORD 'passxyz123');
```

MICHAEL

Especifica o ID de autorização local que você está mapeando para o ID do usuário e senha remotos, que estão definidos no servidor Teradata.

SERVER *tera_server*

Especifica o nome de definição do servidor que você registrou na instrução CREATE SERVER para o servidor Teradata.

REMOTE_AUTHID '*mike*'

Especifica o ID do usuário remoto do Teradata para o qual você está mapeando *MICHAEL*. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_ID para 'U' ou 'L' na instrução CREATE SERVER.

Embora o ID do usuário remoto esteja especificado como uma opção na instrução CREATE SERVER, ele é requerido para as origens de dados Teradata.

REMOTE_PASSWORD '*passxyz123*'

Especifica a senha remota do Teradata que está associada com '*mike*'. O valor faz distinção entre maiúsculas e minúsculas, a menos que você configure a opção do servidor FOLD_PW para 'U' ou 'L' na instrução CREATE SERVER.

Embora a senha remota esteja especificada como uma opção na instrução CREATE SERVER, ela é requerida para as origens de dados Teradata.

USER de Registro Especial do DB2

Você pode utilizar o USER de registro especial do DB2 para mapear o ID de autorização da pessoa que está emitindo a instrução CREATE USER MAPPING para o ID de autorização da origem de dados especificado na opção de usuário REMOTE_AUTHID.

A seguir é apresentado um exemplo da instrução CREATE USER MAPPING a qual inclui o USER de registro especial:

```
CREATE USER MAPPING FOR USER SERVER tera_server
  OPTIONS (REMOTE_AUTHID 'mike', REMOTE_PASSWORD 'passxyz123');
```

Testando a Conexão com o Servidor Teradata

Teste a conexão com o servidor de origem de dados Teradata para determinar se o servidor federado está corretamente configurado para acessar origens de dados Teradata.

Sobre Esta Tarefa

Você pode testar a conexão com o servidor Teradata utilizando a definição de servidor e os mapeamentos de usuário definidos.

Procedimento

Abra uma sessão de passagem e emita uma instrução `SELECT` nas tabelas de sistema Teradata. Se a instrução `SELECT` retornar uma contagem, a definição do servidor e o mapeamento de usuário estão configurados apropriadamente.

Por exemplo:

```
SET PASSTHRU server_definition_name
SELECT count(*) FROM dbc.tables
SET PASSTHRU RESET
```

Se a instrução `SELECT` retornar um erro, você deverá resolver os problemas dos erros de conexão.

O que Fazer Depois

Após concluir esta tarefa, você pode registrar os apelidos para tabelas e visualizações do Teradata.

Resolução de Problemas de Erros de Conexão da Origem de Dados

Uma conexão de teste com o servidor de origem de dados pode retornar um erro por diversas razões. Há ações que você pode executar para determinar a causa de um erro.

Sintoma

É retornado um erro quando você tentar se conectar à origem de dados.

Causa

Há diversas causas possíveis para um problema de conexão.

Solução do Problema

Para resolver problemas de erros de conexão com a origem de dados, verifique os seguintes itens:

- Verifique se a origem de dados está disponível.
- Se aplicável, certifique-se de que o servidor de origem de dados esteja configurado para conexões de chegada.
- Certifique-se de suas configurações de mapeamento de usuário para as opções `REMOTE_AUTHID` e `REMOTE_PASSWORD` sejam válidas para as conexões com a origem de dados. Altere o mapeamento de usuário ou crie outro, conforme necessário.

- Se aplicável, certifique-se de que o software do cliente da origem de dados no servidor federado esteja instalado e configurado corretamente para conexão com a origem de dados.
- Para origens de dados ODBC, certifique-se de que o driver ODBC no servidor federado esteja instalado e configurado corretamente para conexão com o servidor de origem de dados ODBC. Em servidores federados que executam Windows, utilize a ferramenta Administrador de Origem de Dados ODBC para verificar o driver. Em servidores de dados que executam UNIX, consulte a documentação do fornecedor do cliente ODBC.
- Verifique se as configurações para as variáveis configuradas no servidor federado estão corretas para a origem de dados. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis no arquivo `db2dj.ini` e as variáveis do DB2 Profile Registry (`db2set`).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e a crie novamente.

Registrando Apelidos para as Tabelas e Visualizações do Teradata

Para cada definição de servidor Teradata que você registrar, deve registrar um apelido para cada tabela ou visualização que deseja acessar. Utilize estes apelidos, ao invés dos nomes dos objetos de origem de dados, quando consultar os servidores Teradata.

Antes de Iniciar

O banco de dados federado confia nas estatísticas do catálogo da origem de dados para otimizar o processamento da consulta. Para garantir que o banco de dados federado possua estatísticas completas nas tabelas do Teradata, utilize o comando **COLLECT STATISTICS** do Teradata antes de registrar um apelido.

No servidor Teradata, utilize o comando **COLLECT STATISTICS** do Teradata para coletar estatísticas em uma ou mais colunas ou índices em uma tabela.

Ao registrar o apelido com a instrução **CREATE NICKNAME**, o banco de dados federado lê as estatísticas no catálogo do sistema do Teradata e atualiza as estatísticas locais para o apelido.

Sobre Esta Tarefa

Ao registrar um apelido em uma visualização do Teradata, o banco de dados federado reconhece todas as colunas da visualização como anuláveis, mesmo que as colunas na visualização do Teradata não permitam valores nulos. Não há uma solução alternativa para esta limitação.

Procedimento

Emita a instrução **CRIAR APELIDO** a partir da linha de comandos.

Por exemplo:

```
CREATE NICKNAME nickname
FOR server_definition_name.remote_schema.remote_table;
```

Os apelidos podem ter até 128 caracteres de comprimento.

Ao criar o apelido, o servidor federado consulta o catálogo de origem de dados

utilizando o apelido. Esta consulta testa a conexão com a tabela ou visualização da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

O que Fazer Depois

Repita esta etapa para cada tabela ou visualização do Teradata para a qual deseja criar um apelido.

Apelidos do Teradata em Servidores Federados

Ao consultar uma origem de dados Teradata a partir de um servidor federado, você utiliza um apelido na consulta para identificar a tabela e a visualização do Teradata que deseja acessar.

Ao criar um apelido para uma tabela ou visualização do Teradata, o servidor federado conecta-se ao servidor Teradata associado com a tabela ou visualização. O servidor federado utiliza o apelido para verificar a conexão com o servidor Teradata. O banco de dados federado verifica a presença da tabela ou visualização na origem de dados e então, tenta reunir dados estatísticos sobre a tabela ou visualização do Teradata a partir do catálogo no servidor Teradata. As estatísticas que são reunidas sobre o objeto que recebeu o apelido são armazenadas no catálogo global no servidor federado.

O servidor federado utiliza as estatísticas para os objetos que receberam apelidos que ele coleta para otimizar o processamento de consultas. Como algumas ou todas as informações do catálogo do Teradata podem ser utilizadas pelo otimizador de consultas, você deve atualizar as estatísticas no servidor Teradata antes de criar um apelido. Atualize as estatísticas no servidor Teradata utilizando um comando ou utilitário que seja equivalente ao comando DB2 RUNSTATS.

Você não pode enviar uma instrução INSERT, UPDATE ou DELETE para um apelido que faça referência a uma visualização atualizável do Teradata, a menos que a instrução SQL possa ser completamente pushed down para a origem de dados do Teradata.

Instrução CREATE NICKNAME - Exemplos para o Wrapper Teradata

Utilize a instrução CREATE NICKNAME para registrar um apelido para uma tabela ou visualização do Teradata que você deseja acessar. Este tópico inclui um exemplo completo com os parâmetros requeridos.

Este exemplo mostra como criar um apelido para uma tabela ou visualização do Teradata no servidor Teradata:

```
CREATE NICKNAME TERASALES FOR tera_server.vinnie.europe ;
```

TERASALES

Um apelido exclusivo que é utilizado para identificar a tabela ou visualização do Teradata.

Importante: O apelido é um nome de duas partes -o esquema e o apelido. Se você omitir o esquema ao registrar o apelido, o esquema do apelido será o ID de autorização do usuário que registrar o apelido. O ID de autorização é para o servidor federado, não para a origem de dados remota do Teradata.

tera_server.vinnie.europe

Um identificador de três partes para o objeto remoto:

- *tera_server* é o nome da definição de servidor que você designou ao servidor de banco de dados Teradata na instrução CREATE SERVER.
- *vinnie* é o ID do usuário do proprietário ao qual pertence a tabela ou visualização. Este valor faz distinção entre maiúsculas e minúsculas.
- *europe* é o nome da tabela ou visualização remota que você deseja acessar.

Resolução de Problemas da Configuração da Origem de Dados do Teradata

Ativando os Links de Tempo de Execução para libcliv2.so (AIX)

Se você executar o arquivo `djxlinkTeradata.sh` para estabelecer um link com a biblioteca compartilhada do Teradata denominada `libcliv2.so`, pode receber uma mensagem de erro quando emitir uma instrução CREATE NICKNAME.

Sintoma

Um exemplo de uma mensagem de erro que você pode receber é:

```
DB21034E O comando foi processado como uma instrução SQL pois ele não era um
comando do Processador de Linha de Comandos válido. Durante o processamento
da SQL, ele retornou:
SQL30081N Foi detectado um erro de comunicação. Protocolo de comunicação
sendo utilizado: "TCP/IP". API de comunicação sendo utilizada: "SOCKETS". Posição
em que o erro foi detectado: "9.112.26.28". Função de
comunicação detectando
o erro: "recv". Código(s) de erro específico(s) do protocolo: "*", "*", "0".
SQLSTATE=08001
```

Causa

A função `OsCall` fez com que o servidor federado fosse interrompido.

Solução do Problema

Se você receber uma mensagem de erro, verifique o diretório `/sqllib/db2dump` para saber sobre qualquer arquivo trap. Os nomes de arquivo trap começam com a letra `t` e são encerrados com um sufixo `000`. Por exemplo:

```
t123456.000
```

Verifique as informações de rastreamento no arquivo trap para obter qualquer referência à função `OsCall` que indique a função `OsCall` fez com que o servidor federado fosse interrompido.

O exemplo a seguir mostra informações de rastreamento com uma referência à função `OsCall` que você pode localizar em um arquivo trap:

***** Iniciar análise retrospectiva de pilha *****

```
0x239690E0 OsCall + 0x28C
0x23973FB0 mtdpassn + 0x8A4
0x239795A4 mtdp + 0x208
0x2395A928 MTDPIO + 0x28C
0x239609C4 CLICON + 0xD50
0x23962350 DBCHCL + 0xC4
```

Se você localizar uma referência à função `OsCall` em um dos arquivos trap, deve ativar o link de tempo de execução com a biblioteca compartilhada `libcliv2.so`.

Emita os seguintes comandos UNIX para ativar os links de tempo de execução:

```
cd /usr/lib
rtl_enable libcliv2.so -F libtli.a
mv libcliv2.so libcliv2.so.old
mv libcliv2.so.new libcliv2.so
chmod a+r libcliv2.so
```

Resolução de Problemas de Conjuntos de Caracteres para Origens de Dados Teradata

Ao configurar a variável de ambiente `TERADATA_CHARSET` para uma origem de dados Teradata, você pode encontrar erros se o conjunto de caracteres correto não estiver especificado.

Sintoma

Se o conjunto de caracteres correto não estiver especificado para a origem de dados Teradata, você encontrará o seguinte erro:

```
SQL 1822N Código de erro inesperado "227"
recebido da origem de dados "<string>". O texto e os
tokens associados são "MTDP: EM_CHARNAME(227): nome de conjunto de caracteres
inválidos especificado". SQLSTATE=560BD
```

Causa

O conjunto de caracteres especificado na variável de ambiente `TERADATA_CHARSET` está incorreto.

Solução do Problema

Verifique se o conjunto de caracteres correto está instalado e especificado no servidor Teradata e no arquivo `db2dj.ini`.

Resolução de Problemas de Erros na Operação UPDATE ou DELETE em Apelidos

Por padrão, linhas não são identificadas exclusivamente nas tabelas da origem de dados Teradata. Você pode receber uma mensagem de erro SQL ao tentar atualizar ou excluir um apelido que está associado com uma tabela ou visualização do Teradata.

Sintoma

O erro `SQL30090N, RC="21"` é retornado quando você tenta atualizar ou excluir um apelido que está associado com uma visualização ou tabela do Teradata.

Causa

Este problema é causado pois as linhas não são definidas exclusivamente na tabela de origem Teradata.

Solução do Problema

Para solucionar o problema:

1. Elimine e recrie o apelido.
2. Crie pelo menos um índice exclusivo na tabela do Teradata que esteja sendo atualizada ou excluída e tente a operação novamente.

Trabalhando com Criação de Logs de Acesso Teradata

O produto Teradata fornece um recurso de criação de logs de acesso que gera entradas de log quando o Teradata verifica os privilégios de segurança específicos de diversos usuários em um ou mais bancos de dados. Embora a criação de logs de acesso forneça informações de segurança consideráveis e significativas, este recurso aumenta significativamente o uso do processador e pode degradar o desempenho do sistema.

Se você precisar aprimorar o desempenho do sistema, avalie a verificação das regras de privilégios definidas para a criação de logs de acesso. A seguir, encerre qualquer regra desnecessária definindo instruções END LOGGING.

Para melhor desempenho, desative todas as criações de logs de acesso. Elimine o macro Teradata `DBC.AccLogRules` e force uma reconfiguração TPA (Trusted Parallel Application) para parar a criação de logs completamente.

Consulte a documentação do Teradata para obter informações adicionais.

Configurando o Acesso às Origens de Dados de Serviços da Web

Para configurar o sistema federado para acessar origens de dados de serviços da Web, você deve fornecer ao servidor federado informações sobre as origens de dados e objetos que deseja acessar, tal como um documento WSDL (Web Services Description Language) válido.

Antes de Iniciar

- A federação deve ser instalada em um servidor que atuará como o servidor federado.
- Um banco de dados deve existir no servidor federado.

Procedimento

1. Registre o wrapper de serviços da Web.
2. Registre a definição de servidor para as origens de dados de serviços da Web.
3. Registre os mapeamentos de usuários para ativar a segurança para a autenticação HTTP (opcional)
4. Registrar apelidos para origem de dados de serviços da web.
5. Crie visualizações federadas para apelidos de serviços da Web.

Serviços da Web e o Wrapper de Serviços da Web

Os fornecedores de serviços da Web são descritos pelos documentos WSDL (Web Services Description Language). Você pode utilizar o wrapper de serviços da Web para acessar fornecedores de serviços da Web.

O diagrama da Figura 12 na página 220 mostra a arquitetura dos serviços da Web.

1. Um fornecedor de serviços da Web implementa um serviço e publica as informações WSDL em um intermediário de serviço, tal como um UDDI.
2. O consumidor de serviços pode então utilizar o intermediário de serviço para localizar um fornecedor de serviços da Web.
3. Quando o consumidor de serviços localiza um fornecedor de serviços da Web, ele estabelece uma ligação com o fornecedor de serviços para poder utilizar o serviço da Web.

4. O consumidor chama o serviço ao trocar mensagens SOAP (Simple Object Access Protocol) entre o solicitante e o fornecedor.

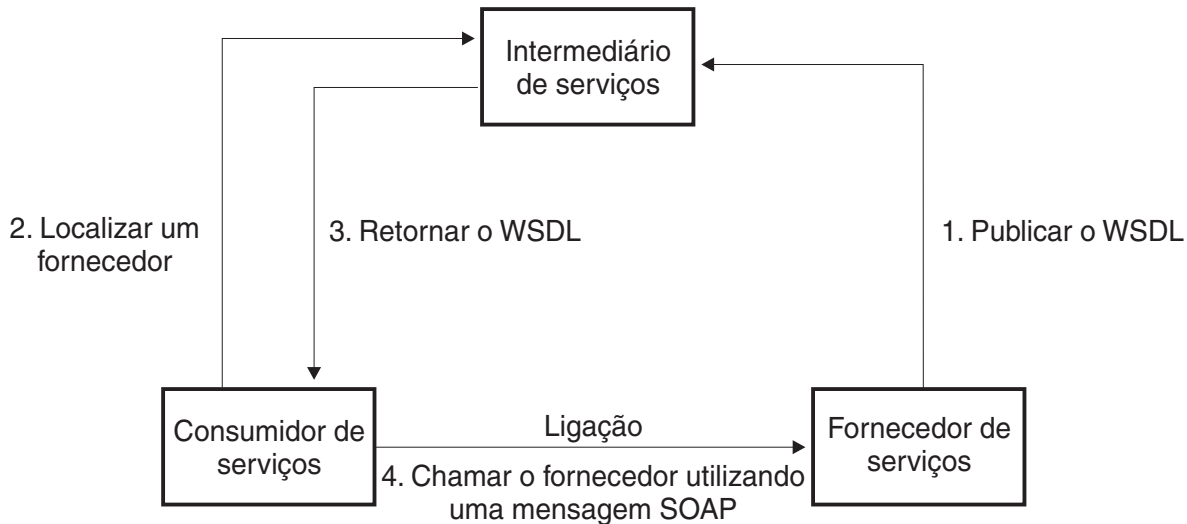


Figura 12. Serviços da Web: Uma Arquitetura Orientada a Serviços

A especificação SOAP define o layout de uma mensagem baseada em XML. Uma mensagem SOAP está contida em um envelope SOAP. O envelope consiste em um cabeçalho SOAP opcional e um corpo SOAP obrigatório. O cabeçalho SOAP pode conter informações sobre a mensagem, tais como informações sobre criptografia ou informações sobre autenticação. O corpo do SOAP contém a mensagem. A especificação SOAP também define uma codificação padrão para ligações de linguagem de programação, que é denominada codificação SOAP.

O Documento WSDL e o Serviço da Web

A chave para o serviço da Web é o documento WSDL. O documento WSDL é um documento XML que descreve os serviços da Web em termos de mensagens que ele envia e recebe. As mensagens são descritas utilizando um sistema de tipos, que, normalmente, é o esquema XML. Uma operação de serviço da Web associa um padrão de troca de mensagens com uma ou mais mensagens. Um padrão de troca de mensagens identifica a seqüência e a cardinalidade das mensagens que são enviadas ou recebidas, bem como para quem e de quem as mensagens são logicamente enviadas ou recebidas. Um interface agrupa operações sem nenhum comprometimento com o formato de transporte ou de conexão. Uma ligação WSDL especifica os detalhes sobre formato de transporte e conexão para uma ou mais interfaces. Um terminal associa um endereço de rede com uma ligação. Um serviço agrupa terminais que implementam uma interface comum. As mensagens podem conter informações orientadas a documentos ou informações orientadas a processos, que também são conhecidas como RPCs (Chamada de Procedimento Remoto). Um documento WSDL pode conter um ou mais serviços da Web.

O exemplo na Figura 13 na página 222 mostra a definição WSDL de um serviço simples que fornece cotações de ações. O serviço da Web suporta uma única operação que é denominada `GetLastTradePrice`. O serviço pode ser acessado com o protocolo SOAP 1.1 sobre HTTP. O pedido lê um símbolo de registrador de cotações como entrada, que é um tipo de dados string, e retorna o preço, que é um tipo de dados float. Os tipos de dados string e float são tipos predefinidos nos padrões de esquema XML. Um serviço da Web também pode definir tipos de

dados e utilizar tais tipos de dados definidos pelo usuário nas mensagens. Os tipos de dados XML predefinidos e definidos pelo usuário são mapeado para as colunas de apelidos. O exemplo completo e a especificação WSDL estão no Web site do W3C.

```

<?xml version="1.0"?>
<definitions name="StockQuote"
...

<types>
  <schema targetNamespace="http://example.com/stockquote.xsd"
    xmlns="http://www.w3.org/2000/10/XMLSchema">
    <element name="TradePriceRequest">
      <complexType>
        <all>
          <element name="tickerSymbol" type="string"/>
        </all>
      </complexType>
    </element>
    <element name="TradePrice">
      <complexType>
        <all>
          <element name="price" type="float"/>
        </all>
      </complexType>
    </element>
  </schema>
</types>

<message name="GetLastTradePriceInput">
...
</message>

  <portType name="StockQuotePortType">
    <operation name="GetLastTradePrice">
      <input message="tns:GetLastTradePriceInput"/>
      <output message="tns:GetLastTradePriceOutput"/>
    </operation>
  </portType>

  <binding name="StockQuoteSoapBinding"
    type="tns:StockQuotePortType">
    <soap:binding style="document"
      transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="GetLastTradePrice">
      <soap:operation soapAction="http://example.com/GetLastTradePrice"/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
    </operation>
  </binding>

  <service name="StockQuoteService">
    <documentation>My first service</documentation>
    <port name="StockQuotePort" binding="tns:StockQuoteBinding">
      <soap:address location="http://example.com/stockquote"/>
    </port>
  </service>
</definitions>

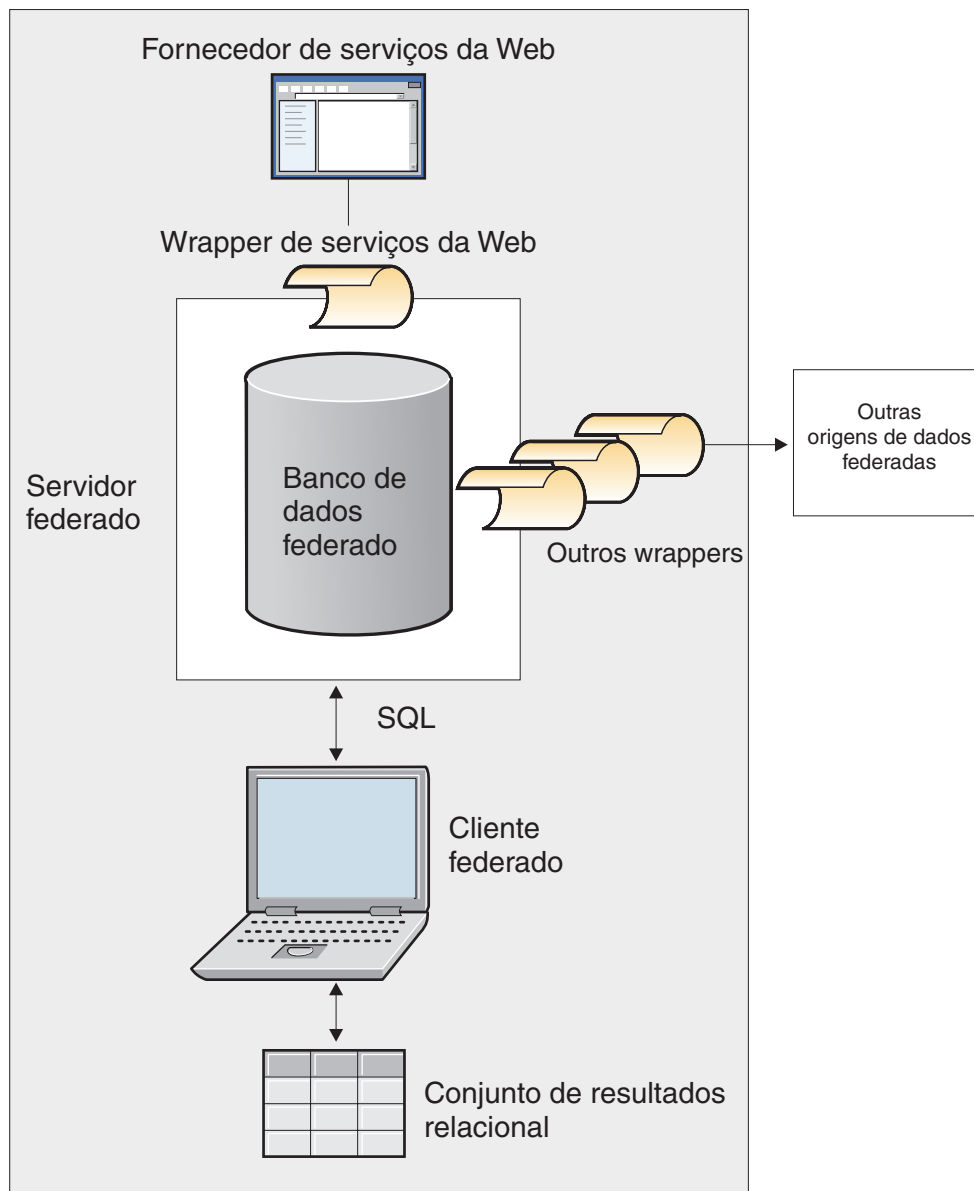
```

Figura 13. Exemplo de um Documento WSDL

O Documento WSDL, o Wrapper de Serviços da Web e os Apelidos

O wrapper de serviços da Web utiliza as operações em um tipo de porta que possui uma ligação SOAP com um transporte HTTP. As mensagens de entrada na operação e os tipos ou elementos associados se transformam em colunas no apelido. As mensagens de saída na operação são extraídas da hierarquia de apelidos. Você pode criar uma hierarquia separada de apelidos para cada operação no documento WSDL.

Ao utilizar o wrapper de serviços da Web, você pode utilizar as funções de sistemas federados para unir dados dos serviços da Web com dados em outras origens de dados federadas.



O exemplo da Figura 14 na página 224 utiliza um documento WSDL que contém um portType com um nome de operação GETTEMP. Com este serviço da Web, você digita um código postal com entrada e recebe uma temperatura para tal

código postal.

```
<?xml version="1.0"?>
<definitions name="TemperatureService" targetNamespace=http://www.xmethods.net/
  sd/TemperatureService.wsdl "
  xmlns:tns="http://www.xmethods.net/sd/TemperatureService.wsdl "
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns="http://schemas.xmlsoap.org/wsdl/">
  <message name="getTempRequest">
    <part name="zipcode" type="xsd:string"/>
  </message>
  <message name="getTempResponse">
    <part name="return" type="xsd:float"/>
  </message>
  <portType name="TemperaturePortType">
    <operation name="getTemp">
      <input message="tns:getTempRequest"/>
      <output message="tns:getTempResponse"/>
    </operation>
  </portType>
  <binding name="TemperatureBinding" type="tns:TemperaturePortType">
    <soap:binding style="rpc"
      transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="getTemp">
      <soap:operation soapAction="" />
      <input>
        <soap:body use="encoded" namespace="urn:xmethods-Temperature"
          encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
      </input>
      <output>
        <soap:body use="encoded" namespace="urn:xmethods-Temperature"
          encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
      </output>
    </operation>
  </binding>
  <service name="TemperatureService">
    <documentation>
      Returns current temperature in a given U.S. zipcode
    </documentation>
    <port name="TemperaturePort" binding="tns:TemperatureBinding">
      <soap:address
        location="http://services.xmethods.net:80/soap/servlet/rpcrouter" />
    </port>
  </service>
</definitions>
```

Figura 14. Serviço da Web GETTEMP

O valor de entrada está descrito na coluna zipcode do apelido. O valor de saída está descrito pela coluna return do apelido. No documento WSDL, tais colunas estão identificadas no elemento messages. O elemento messages representa a definição lógica dos dados que são enviados entre o o fornecedor de serviços da Web e o consumidor de serviços da Web. Se forem necessárias mais explicações sobre as informações no elemento message, então o documento WSDL também pode conter um elemento type. O elemento type pode fazer referência a tipos predefinidos que são baseados nas especificações do esquema XML ou tipos que são definidos por um usuário.

O exemplo Figura 15 na página 225 mostra o apelido produzido a partir do documento WSDL do serviço da web GETTEMP. A coluna zipcode é uma coluna

de entrada requerida devido à sintaxe do apelido TEMPLATE:

```
CREATE NICKNAME GETTEMP (
  ZIPCODE VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  RETURN VARCHAR (48) OPTIONS(XPATH './return/text()')
)
FOR SERVER "EHPWSSERV"
OPTIONS(URL 'http://services.xmethods.net:80/soap/servlet/rpcrouter',
  SOAPACTION ' ',
  TEMPLATE '<soapenv:Envelope>
    <soapenv:Body>
      <ns2:getTemp>
        <zipcode>&zipcode[1,1]</zipcode>
      </ns2:getTemp>
    </soapenv:Body>
  </soapenv:Envelope>',
  XPATH '/soapenv:Envelope/soapenv:Body/*',
  NAMESPACES ' ns1="http://www.xmethods.net/sd/TemperatureService.wsdl",
              ns2="urn:xmethods-Temperature" ,
              soapenv="http://schemas.xmlsoap.org/soap/envelope/" ');
```

Figura 15. Apelido GETTEMP

As opções de apelido no wrapper de serviços da Web, URL e SOAPACTION, fornecem a habilidade para substituir o terminal, ou o endereço que você especificou ao criar o apelido. Quando você utiliza as colunas ativadas URLCOLUMN ou SOAPACTIONCOLUMN em uma consulta, pode utilizar endereços dinâmicos com os mesmos apelidos. Se você definir as opções URL e SOAPACTION do apelido ao criar um apelido e ativar URLCOLUMN e SOAPACTIONCOLUMN na opção da coluna, então você estará utilizando as funções de ligação tardias dos wrappers de serviços Web. O valor para a opção de apelido SOAPACTION transforma-se em um atributo no cabeçalho HTTP. O valor para a opção de apelido URL é a URL HTTP para a qual o pedido é enviado.

As opções de apelido URL e SOAPACTION fornecem associações dinâmicas de apelidos. Estes endereços dinâmicos são úteis se diversas empresas implementam um portType de serviço da Web. O wrapper de serviços da Web requer que as únicas diferenças entre os documentos WSDL sejam URLs e SOAPACTIONS diferentes. Você pode utilizar a função de ligação tardia para criar e utilizar o mesmo apelido para diferentes terminais de serviço que diferentes empresas podem desejar utilizar. Os valores URL e SOAPACTION são derivados do documento WSDL.

O exemplo a seguir mostra como você pode utilizar as opções de coluna URLCOLUMN e SOAPACTIONCOLUMN:

```

CREATE NICKNAME GetPartQuote(
  partnumber INTEGER OPTIONS (TEMPLATE '&column'),
  price FLOAT OPTIONS (XPATH './price'),
  urlcol VARCHAR(100) OPTIONS (URLCOLUMN 'Y'),
  soapactioncol VARCHAR(100) OPTIONS (SOAPACTIONCOLUMN 'Y'),
  FOR SERVER myServer
  OPTIONS (
    ...
    SOAPACTION 'http://example.com/GetPartPrice' ,
    URL 'http://mycompany.com:9080/GetPartPrice',
    ...
  )

```

Figura 16. Apelido GetPartQuote

O exemplo a seguir utiliza as colunas URLCOL e SOAPACTIONCOL que foram definidas com a opção de coluna URLCOLUMN ativada e a opção de coluna SOAPACTIONCOLUMN ativada:

```

SELECT * FROM supplier_endpoints p,
  GetPartQuote q
WHERE partnumber=1234 AND
  p.url=q.urlcol AND
  p.soapaction=q.soapactioncol;

```

O aplicativo SQL pode adiar a escolha de quais terminais utilizar até o momento em que uma consulta é executada, ao invés de definir um terminal específico no momento em que o apelido é criado.

O wrapper de serviços da Web pode separar uma grande quantidade de dados do documento WSDL em fragmentos para diminuir o total de memória que é utilizado. Especifique a opção **STREAMING** ao criar um apelido de serviços da web. O wrapper de serviços da Web processa o fluxo resultante dos dados XML e, então, extrai as informações que são solicitadas por um fragmento de consulta. O wrapper de serviços da Web analisa um segmento a cada vez. Utilize a opção **STREAMING** para analisar apenas grandes documentos XML.

Registrando o Wrapper de Serviços da Web

Você deve registrar um wrapper para acessar origens de dados de serviços da Web. Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Antes de Iniciar

Consulte a lista dos arquivos da biblioteca do wrapper de serviços da Web para obter o nome correto para especificar na instrução CREATE WRAPPER.

Sobre Esta Tarefa

O nome do arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado.

Procedimento

Emita um comando CREATE WRAPPER com o nome do wrapper e o nome do arquivo da biblioteca do wrapper. Por exemplo, para registrar um wrapper com o nome websr_wrapper em um servidor federado que utiliza o Windows, emita a seguinte instrução:

```
CREATE WRAPPER websr_wrapper LIBRARY 'db2ws.dll';
```

Arquivos da Biblioteca do Wrapper de Serviços da Web

Os arquivos da biblioteca do wrapper de serviços da Web são incluídos no servidor federado quando você o instala.

Quando você instala o IBM InfoSphere Federation Server, os arquivos de biblioteca são adicionados no caminho do diretório padrão. Por exemplo, se o servidor federado estiver sendo executado no AIX, os arquivos da biblioteca do wrapper incluídos no caminho do diretório são libdb2ws.a, libdb2wsF.a, e libdb2wsU.a. O arquivo da biblioteca do wrapper padrão é libdb2ws.a. Os outros arquivos da biblioteca do wrapper são utilizados com opções de wrapper específicas.

Inclua o parâmetro LIBRARY na instrução CREATE WRAPPER e especifique o nome do arquivo da biblioteca padrão do wrapper.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 39. Locais e Nomes de Arquivo de Bibliotecas para CREATE WRAPPER

Sistema Operacional	Caminho do Diretório	Arquivo da Biblioteca do Wrapper
AIX	/usr/opt/<install_path>/lib32/ /usr/opt/<install_path>/lib64/	libdb2ws.a
Linux	/opt/IBM/db2/<install_path>/lib32 /opt/IBM/db2/<install_path>/lib64	libsb2ws.so
Solaris	/opt/IBM/db2/<install_path>/lib32 /opt/IBM/db2/<install_path>/lib64	libdb2ws.so
Windows	%DB2PATH%\bin	db2ws.dll

- <install_path> é o caminho do diretório em que o servidor federado está instalado no Linux ou UNIX.

%DB2PATH% é a variável de ambiente utilizada para especificar o caminho do diretório onde o servidor federado está instalado no Windows. O caminho padrão do caminho do diretório do Windows é C:\Arquivos de Programa\IBM\SQLLIB.

Registrando a Definição de Servidor para Origens de Dados de Serviços da Web

Você deve registrar uma definição de servidor de serviços da web para cada serviço da web que deseja acessar.

Procedimento

Para registrar uma definição de servidor no sistema federado para o wrapper de serviços da web, emita a instrução CRIAR SERVIDOR a partir da linha de comandos.

Por exemplo, para registrar uma definição de servidor de serviços da Web denominada `ws_server` no Windows, emita a seguinte instrução:

```
CREATE SERVER ws_server WRAPPER websr_wrapper;
```

Você pode configurar parâmetros de tempo limite e de servidor proxy opcionais para a instrução `CREATE SERVER`.

Instrução `CREATE SERVER` - Exemplos para o Wrapper de Serviços da Web

Utilize a instrução `CREATE SERVER` para registrar definições de servidor para o wrapper de serviços da Web com configurações de tempo limite e de servidor proxy.

Mesmo que você não esteja utilizando um servidor proxy para acessar documentos de serviços da Web, ainda será necessário registrar uma definição de servidor. A hierarquia dos objetos federados requer que os arquivos de serviços da Web estejam associados com um objeto de definição de servidor específico. A instrução que você deve emitir para registrar uma definição de servidor é:

```
CREATE SERVER my_server WRAPPER my_wrapper  
    OPTIONS (TIMEOUT '60');
```

my_server

Um nome que você designa para a definição de servidor de serviços da Web. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER my_wrapper

O Nome do Wrapper Especificado na Instrução `CREATE WRAPPER`.

TIMEOUT '60'

Especifica o tempo, em minutos, que o servidor federado deve aguardar por uma transferência de rede e pelo cálculo de um resultado.

Definições de servidor quando um servidor proxy é usado

Use as opções do servidor proxy na instrução `CREATE SERVER` se todas estas condições forem verdadeiras:

- Para recuperar dados usando uma URI.
- A URI recupera dados por trás de um firewall, através de um proxy.
- O firewall ou proxy é HTTP ou SOCKS.

As opções que você especifica dependem do tipo do servidor de proxy que você deseja acessar.

Verifique com o administrador de rede as informações sobre o tipo de proxy que você utiliza, e as configurações que você deve especificar nas opções do proxy.

Registrando uma definição de servidor para um servidor de proxy HTTP

Para registrar uma definição de servidor e especificar um servidor de proxy HTTP, utilize a seguinte instrução:

```
CREATE SERVER ws_server_http  
    WRAPPER ws_wrapper  
    OPTIONS (PROXY_TYPE 'HTTP', PROXY_SERVER_NAME 'proxy_http',  
            PROXY_SERVER_PORT '8080');
```

ws_server_http

Um nome que você designa para a definição de servidor de serviços da Web. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER *ws_wrapper*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

PROXY_TYPE 'HTTP'

Especifica o tipo de proxy que é utilizado para acessar a Internet quando se está por trás de um firewall.

PROXY_SERVER_NAME 'proxy_http'

Especifica o nome do servidor proxy ou o endereço IP.

PROXY_SERVER_PORT '8080'

Especifica o número da porta do servidor proxy.

Registrando uma definição de servidor para um servidor proxy SOCKS com informações sobre autenticação

Para registrar uma definição de servidor e especificar um servidor proxy SOCKS com informações de autenticação, emita esta instrução:

```
CREATE SERVER ws_server_socks
  WRAPPER ws_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS', PROXY_SERVER_NAME 'proxy_socks',
    PROXY_SERVER_PORT '1081', PROXY_AUTHID 'Sampson',
    PROXY_PASSWORD 'sailing4me');
```

PROXY_AUTHID 'Sampson'

Especifica o nome de usuário no servidor de proxy.

PROXY_PASSWORD 'sailing4me'

Especifica a senha no servidor proxy que está associada com o nome do usuário *Sampson*.

Ativando a Segurança através do Wrapper de Serviços da Web

O wrapper de serviços da Web suporta autenticação HTTP ao utilizar a instrução CREATE USER MAPPING.

Sobre Esta Tarefa

O wrapper de serviços da Web suporta HTTPS como um protocolo de transporte para mensagens SOAP. O documento WSDL que é gerado pelo fornecedor de serviços da Web contém https:// na URL. Deste modo, as mensagens SOAP no pedido HTTP ou na resposta HTTP são criptografadas.

Se o serviço da Web utiliza HTTPS como um protocolo de transporte, você pode configurar o wrapper para validar os certificados SSL que o servidor envia para identificação utilizando o wrapper ou a opção de servidor SSL_VERIFY_SERVER_CERTIFICATE. O wrapper de serviço da Web pode chamar serviços da Web com certificados auto-assinados.

O mapeamento de usuário é opcional. Se você não especificar um mapeamento de usuário, poderá visualizar um erro se o fornecedor de serviços da Web esperar informações de autenticação. Alguns servidores podem utilizar autenticação para restringir o acesso a um serviço. A necessidade de autenticação não é aparente nas informações no documento WSDL.

Procedimento

Emita a instrução CREATE USER MAPPING.

Exemplo

Por exemplo, com a instrução CREATE USER MAPPING a seguir, quando o apelido de serviços da Web no servidor S1 é acessado, o pedido HTTP é enviado com SYSTEM como o ID do usuário e MANAGER como a senha.

```
CREATE USER MAPPING
  FOR RSPALTEN SERVER S1
  OPTIONS ( REMOTE_AUTHID 'SYSTEM', REMOTE_PASSWORD 'MANAGER'
          PROXY_AUTHID 'ID' PROXY_PASSWORD 'PWD'
          SSL_CLIENT_CERTIFICATE_LABEL 'LABEL');
```

Registrando Apelidos para as Origens de Dados de Serviços da Web

Para cada definição de servidor de serviços da Web que registrar, você deve registrar um apelido para cada origem de dados que deseja acessar. Utilize estes apelidos ao invés dos nomes das origens de dados ao consultar as origens de dados de serviços da Web.

Antes de Iniciar

Você deve ter acesso a um documento WSDL válido que descreva o serviço da Web com o qual deseja se comunicar.

Sobre Esta Tarefa

Restrições

- Apenas operações de pedido-resposta são suportadas.
- Uma ligação SOAP com um transporte HTTP é a única ligação que é suportada.
- Você deve utilizar a opção TEMPLATE ou a opção XPATH em cada coluna, exceto para as colunas especiais com as opções SOAPACTIONCOLUMN, URLCOLUMN, PRIMARY_KEY ou FOREIGN_KEY.

Você cria uma hierarquia de apelidos para cada operação de serviço da Web que está definida no documento WSDL (Web Services Description Language). O apelido-pai identifica o envelope SOAP; os apelidos-filhos identificam os elementos de corpo SOAP e o cabeçalho SOAP opcional.

Procedimento

Emita uma instrução CRIAR APELIDO a partir da linha de comandos. Por exemplo, para registrar apelidos no Windows para um serviço da Web denominado GETTEMP, emita a seguinte instrução:

```
CREATE NICKNAME GETTEMP (
  ZIPCODE VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  RETURN VARCHAR (48) OPTIONS(XPATH './return/text()')
)
FOR SERVER "EHPWSSERV"
  OPTIONS(URL 'http://services.xmethods.net:80/soap/servlet/rpcrouter',
         SOAPACTION ' ',
         TEMPLATE '<soapenv:Envelope>
                  <soapenv:Body>
                  <ns2:getTemp>
```

```

        <zipcode>&zipcode[1,1]</zipcode>
    </ns2:getTemp>
</soapenv:Body>
</soapenv:Envelope>',
XPATH '/soapenv:Envelope/soapenv:Body/*' ,
NAMESPACES ' ns1="http://www.xmethods.net/sd/TemperatureService.wsdl",
            ns2="urn:xmethods-Temperature" ,
            soapenv="http://schemas.xmlsoap.org/soap/envelope/" ');

```

Instrução CREATE NICKNAME - Exemplos para Wrapper de Serviços da Web

Ao criar um apelido para acessar um serviço da Web, você cria uma coluna de entrada para cada valor na mensagem de entrada de uma operação de serviço da Web e uma coluna de saída para cada valor na mensagem de saída de uma operação de serviço da Web. Você controla as definições de coluna de entrada e saída com as definições de opções de colunas de apelidos.

A opção de coluna TEMPLATE especifica que uma coluna é uma coluna de entrada. A opção de coluna XPATH especifica que uma coluna é uma coluna de saída. Quando a opção de apelido TEMPLATE contém uma notação entre colchetes ([1,1]), a coluna é uma coluna de entrada requerida.

A opção de apelido NAMESPACES é uma lista de pares nome-valor separados por vírgulas que um sistema federado utiliza para resolver os espaços de nomes para elementos em documentos XML de entrada e saída. Os nomes de espaços são utilizados no pedido de mensagem para que os prefixos na opção de apelido TEMPLATE sejam definidos. A opção de apelido NAMESPACES resolve os prefixos nas expressões XPath com as URIs de espaço de nomes que são definidas no WSDL ou nos esquemas XML. As expressões XPath são aplicadas no documento XML que é retornado do serviço da Web.

Exemplo 1: Colunas de Entrada Requeridas

O exemplo a seguir mostra um apelido para um serviço da Web denominado getQuote. O serviço da Web lê um símbolo de registrador de cotações de ações e retorna um preço de comercial.

```

CREATE NICKNAME "stockquote.stockquoteport_getquote_nn" (
    symbol VARCHAR (48) OPTIONS(TEMPLATE '&column'),
    result VARCHAR (48) OPTIONS(XPATH './Result/text()'))
FOR SERVER "xmethods_server" OPTIONS(
    URL 'http://66.28.98.121:9090/soap' ,
    SOAPACTION 'urn:xmethods-delayed-quotes#getQuote' ,
    TEMPLATE '<soapenv:Envelope>
        <soapenv:Body>
            <ns2:getQuote>
                <symbol>&symbol[1,1]</symbol>
            </ns2:getQuote>
        </soapenv:Body>
    </soapenv:Envelope>',
    XPATH '/soapenv:Envelope/soapenv:Body/*' ,
    NAMESPACES 'ns2="urn:xmethods-delayed-quotes" ,
                ns1="http://www.example.com/wsdl/
                net.xmethods.services.stockquote.StockQuote/" ,
                soapenv="http://schemas.xmlsoap.org/soap/envelope/" ');

```

A opção TEMPLATE do apelido especifica a coluna SYMBOL como uma coluna de entrada requerida, pois a coluna contém a designação [1,1]. Na opção TEMPLATE do apelido, o envelope SOAP completo é especificado para o serviço da Web. O valor de entrada getQuote está contido no envelope SOAP e nos elementos de

corpo. A opção de apelido XPATH contém as informações para localizar o preço comercial através do envelope SOAP e das tags de corpo.

Utilize o apelido stockquote.stockquoteport_getquote_nn para acessar o serviço da Web, tal como na consulta a seguir:

```
SELECT * FROM "stockquote.stockquoteport_getquote_nn"  
WHERE symbol='IBM';
```

Você deve utilizar o predicado, `symbol='IBM'`, nesta instrução pois `symbol` é uma coluna de entrada requerida. O predicado de igualdade é o único predicado válido nas colunas de entrada. Cada um dos predicados de igualdade configura um valor na mensagem de entrada. Se a coluna de entrada for opcional, o predicado de igualdade nesta coluna não será necessário. Se a coluna de entrada for requerida, então, você deve emitir a consulta com um predicado de igualdade. Você pode utilizar um valor literal tal como `IBM` em uma expressão de igualdade ou um valor de uma tabela unida ou de um apelido.

Exemplo 2: Repetindo Elementos e Apelidos-filhos

O exemplo a seguir utiliza um serviço da Web denominado `getZooReport` que produz um relatório para zoológicos. O valor de entrada é um identificador do zoológico. O valor de saída é um relatório que é descrito pelo seguinte esquema:

```

<wsdl:definitions name="Name"
  targetNamespace="http://myzoo.com"
  ...
<wsdl:types>
  <xsd:schema elementFormDefault="qualified" targetNamespace="http://myzoo.com"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:element name="Animal">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:Name"/>
          <xsd:element ref="tns:Species"/>
          <xsd:element ref="tns:Lot"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="AnimalCareList">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element maxOccurs="unbounded" minOccurs="1" ref="tns:Animal"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="Count" type="xsd:string"/>
    <xsd:element name="LastModified" type="xsd:string"/>
    <xsd:element name="Lot" type="xsd:string"/>
    <xsd:element name="Name" type="xsd:string"/>
    <xsd:element name="NumberOfCages" type="xsd:string"/>
    <xsd:element name="Species" type="xsd:string"/>
    <xsd:element name="Zoo">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:ZooName"/>
          <xsd:element ref="tns:Count"/>
          <xsd:element ref="tns:LastModified"/>
          <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:Zookeeper"/>
        </xsd:sequence>
        <xsd:attribute name="id" type="xsd:string" use="optional"/>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="ZooName" type="xsd:string"/>
    <xsd:element name="Zookeeper">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:Name"/>
          <xsd:element ref="tns:NumberOfCages"/>
          <xsd:element ref="tns:AnimalCareList"/>
        </xsd:sequence>
        <xsd:attribute name="id" type="xsd:string" use="optional"/>
      </xsd:complexType>
    </xsd:element>
  </xsd:schema>
</wsdl:types>
  ...

```

Figura 17. Serviço da Web *getZooReport*

A seguinte instrução DDL é baseada no WSDL que contém o esquema:

```

CREATE NICKNAME zooport_getzooreport_nn (
  zoid VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  zoo_id VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/@ns1:id'),
  report_zooname VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/ns1:ZooName/text()'),
  report_count VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/ns1:Count/text()'),
  report_lastmodified VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:LastModified/text()'),
  zooport_getzooreport_pkey VARCHAR (16) FOR BIT DATA NOT NULL
    OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER "zooserver" OPTIONS(
  URL 'http://localhost:9080/MaelstromTest/services/ZooPort' ,
  SOAPACTION 'http://myzoo.com/getZooReport' ,
  TEMPLATE '<soapenv:Envelope>
    <soapenv:Body>
      <zooId>&zooId[1,1]</zooId>
    </soapenv:Body>
  </soapenv:Envelope>',
  XPATH '/soapenv:Envelope/soapenv:Body' ,
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
              ns1="http://myzoo.com " ');

```

Figura 18. Apelido do pai - zooport_getzooreport_nn

```

CREATE NICKNAME zooport_getzooreport_report_nn (
  zooport_getzooreport_fkey VARCHAR (16)
    FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'ZOOPORT_GETZOOREPORT_NN'),
  zookeeper_id VARCHAR (48) OPTIONS(XPATH './ns1:Zookeeper/@ns1:id'),
  report_name VARCHAR (48) OPTIONS(XPATH './ns1:Zookeeper/ns1:Name/text()'),
  report_numberofcages VARCHAR (48)
    OPTIONS(XPATH './ns1:Zookeeper/ns1:NumberOfCages/text()'),
  zooport_getzooreport_pkey VARCHAR (16)
    FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER "zooserver" OPTIONS(
  XPATH './ns1:Zoo' ,
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
              ns1="http://myzoo.com" ');

```

Figura 19. Filho do Apelido zooport_getzooreport_nn

```

CREATE NICKNAME zooport_getzooreport_report_report_nn (
  zooport_getzooreport_fkey VARCHAR (16) FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'zooport_getzooreport_report_nn'),
  report_name VARCHAR (48)
    OPTIONS(XPATH './ns1:Animal/ns1:Name/text()'),
  report_species VARCHAR (48)
    OPTIONS(XPATH './ns1:Animal/ns1:Species/text()'),
  report_lot VARCHAR (48) OPTIONS(XPATH './ns1:Animal/ns1:Lot/text()'))
FOR SERVER "zooserver" OPTIONS(
  XPATH './ns1:Zookeeper/ns1:AnimalCareList' ,
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
              ns1="http://myzoo.com" ');

```

Figura 20. Filho de zooport_getzooreport_report_nn

O esquema inclui alguns elementos que são repetidos, conhecidos como elementos de seqüência. Estes elementos se tornam apelidos-filhos do apelido-pai, conforme mostrado na Figura 18, Figura 19 e Figura 20. Por exemplo, zooname, count,

lastmodified e zookeeper são todos elementos de zoo. O elemento zoo contém 0 ou mais elementos zookeeper. O apelido raiz, zoo, contém as colunas zooname, count e lastmodified. Um apelido filho, zookeeper, é criado para descrever os elementos de repetição de zookeeper. O terceiro elemento na coluna zookeeper, animalcarelist, também contém 0 ou mais elementos e, portanto, ele se torna um apelido-filho, zooport_getzooreport_report_report_nn. A figura a seguir mostra a hierarquia de apelidos:

Apelido raiz: zooport_getzooreport_nn

Zoo (pai):

- ZooName
- Count
- LastModified
- ZooKeeper (há 0 ou mais elementos ZooKeeper)

Apelido-filho: zooport_getzooreport_report_nn

Elementos ZooKeeper

- Nome
- NumberOfCages
- AnimalCareList (há 0 ou mais elementos Animal)

Apelido-filho: zooport_getzooreport_report_report_nn

Animal

- Nome
 - Species
 - Lot
-

Figura 21. Hierarquias de apelido de Pai -> Filho ->

A instrução a seguir é uma consulta típica que você pode emitir nos apelidos para acessar o serviço da Web de relatório do zoológico. Quando você emite esta instrução, você recupera as informações do relatório do zoológico com base em um identificador específico e no local onde as chaves primárias e estrangeiras dos relatórios do zoológico do apelido-filho correspondem.

```
SELECT * FROM zooport_getzooreport_nn ,
        zooport_getzooreport_report_nn zk ,
        zooport_getzooreport_report_report_nn a
WHERE zooid='1'AND zooport_getzooreport_pkey=zk.zooport_getzooreport_fkey
and zk.zooport_getzooreport_pkey=a.zooport_getzooreport_fkey;
```

Exemplo 3: Ligação Tardia

O exemplo a seguir mostra como você pode utilizar a opção de ligação tardia. Se você definir as opções de apelido URL e SOAPACTION e se ativar as opções de coluna URLCOLUMN e SOAPACTIONCOLUMN ao criar um apelido, você estará utilizando as funções de ligação tardias.

O exemplo a seguir é para um serviço da Web que fornece cotações de preços para peças que são implementados por todos os fornecedores de uma empresa. A seguir está a instrução CREATE NICKNAME que inclui as definições URLCOLUMN e SOAPACTIONCOLUMN:

```
CREATE NICKNAME GetPartQuote(
    partnumber INTEGER OPTIONS (TEMPLATE'&column'),
    price FLOAT OPTIONS (XPATH './price'),
    urlcol1 VARCHAR(100) OPTIONS (URLCOLUMN 'Y'),
```

```

soapactioncol VARCHAR(100) OPTIONS (SOAPACTIONCOLUMN 'Y'),
FOR SERVER myServer
OPTIONS (
...
SOAPACTION 'http://example.com/GetPartPrice' ,
URL 'http://mycompany.com:9080/GetPartPrice'',
...
)

```

Para obter cotações de preços de todos os fornecedores com uma única consulta, os valores que cada fornecedor utiliza para as opções de coluna SOAPACTION e URL são necessários. valores das colunas como yes. A consulta é semelhante à seguinte:

```

SELECT * FROM supplier_endpoints p,
       GetPartQuote q
WHERE partnumber=1234 AND
       p.url=q.urlcol AND
       p.soapaction=q.soapactioncol;

```

A tabela local `supplier_endpoints` contém todas as URLs e endereços SOAP com os quais você chama o serviço da Web. Você pode incluir uma cláusula de preço `ORDER BY` para determinar o fornecedor com menor preço para esta peça.

Exemplo 4: Opção de Coluna `ESCAPE_INPUT`

Você pode incluir fragmentos XML como valores de entrada em sua consulta. Ao registrar um apelido, inclua a opção de coluna `ESCAPE_INPUT=N`. Esta opção mantém os caracteres especiais, tais como sinais de maior e menor (`<` and `>`) em fragmentos XML nos valores de entrada.

Quando um esquema contém valores de entrada repetidos que requerem que você envie XML como parte da mensagem SOAP, você pode utilizar a opção de coluna `ESCAPE_INPUT` para criar a mensagem de saída com o XML correto.

Por exemplo, o serviço da Web do zoológico inclui uma operação para incluir um novo zelador de animais e os animais que estão associados com este zelador. No esquema para este exemplo, uma `AnimalCareList` pode ter vários animais.

```

CREATE NICKNAME add_zookeeper(
  zookeeper_id VARCHAR(48) OPTIONS(TEMPLATE '...'),
  name VARCHAR(48) OPTIONS(TEMPLATE '...'),
  numberofcages VARCHAR(48) OPTIONS(TEMPLATE '...'),
  animals VARCHAR(3000) OPTIONS( TEMPLATE '...' , ESCAPE_INPUT 'N')
...

```

Para incluir um novo zelador de animais, emita uma consulta semelhante à do exemplo a seguir:

```

SELECT * FROM add_zookeeper
WHERE zookeeper_ID='37' AND
      name='Amit Kapoor' AND
      numberofcages='3' AND
      animals='<AnimalCareList xmlns="http://myzoo.com">
              <Animal>
                <Name>Larry</Name>
                <Species>Gorilla</Species>
                <Lot>7</Lot>
              </Animal>
              <Animal>
                <Name>Bill</Name>
                <Species>Chimpanzee</Species>
                <Lot>8</Lot>
              </Animal>
            </AnimalCareList>';

```

O apelido `add_zookeeper` é uma operação de serviço da Web que pode alterar o estado do serviço da Web ou atualizar as informações. Embora os wrappers não-relacionais não possam ser atualizados, a instrução `SELECT` neste exemplo atualiza as informações do zoológico para incluir um novo zelador de animais.

Você também pode utilizar a opção de coluna `ESCAPE_INPUT` para um esquema com um elemento como `xsd:anyType`. Neste caso, o tipo do elemento é desconhecido. Você pode utilizar a opção de coluna `ESCAPE_INPUT` na coluna de entrada para este elemento para que seja possível especificar fragmentos XML arbitrários para sua entrada.

acesso ao cabeçalho SOAP

Utilize o wrapper de Serviço da Web para acessar valores do cabeçalho SOAP (Simple Object Access Protocol) que estão incorporados em uma mensagem SOAP.

Uma mensagem SOAP, seja um pedido ou uma resposta, consiste em um envelope que contém os nomes de espaço que são utilizados pelo restante da mensagem SOAP, um cabeçalho opcional e um corpo. Um cabeçalho SOAP normalmente contém metadados sobre a mensagem SOAP ou outras informações opcionais. A mensagem SOAP se movimenta de uma origem para um destino passando por uma série de serviços intermediários que processam a mensagem ou seus efeitos secundários. Por exemplo, uma mensagem pode passar por um serviço de transação que fornece ao cliente a chamada garantida no caso de uma falha na rede, ou um serviço de segurança que está localizado em um portal corporativo pode fornecer informações sobre autenticação. Estes intermediários sempre atuam sobre os metadados contidos no cabeçalho SOAP.

Os cabeçalhos SOAP sempre contêm informações opcionais ou suporte ao desenvolvimento de interfaces com o usuário. Por exemplo, um banco possibilita que os clientes utilizem cartões de saque para retirar dinheiro de suas contas correntes. Entretanto, alguns clientes possuem mais de uma conta. Se um cliente possui várias contas, o banco deve ser capaz de identificar a conta da qual foi feita a retirada.

Utilizando este exemplo, a seguir está o elemento do cabeçalho que especifica uma conta bancária:

```
<xsd:element name="AccountID" type="xsd:int"/>
```

A seguir está uma mensagem SOAP que utiliza este elemento do cabeçalho. Neste caso, o cliente selecionou retirar 1.000 dólares da conta 1 e ocorreu uma taxa de transação de 50 cents.

```
<SOAP-ENV:Envelope>
  <SOAP-ENV:Header>
    <tns:AccountID>1</tns:AccountID>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <tns:Action>withdrawal</tns:Action>
    <tns:Amount>1000</tns:Amount>
    <tns:Fee>0.50</tns:Fee>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Os cabeçalhos SOAP possuem estes requisitos:

- `soapenv:Header` é um elemento opcional sob `soapenv:Envelope`.
- Se utilizado, o cabeçalho SOAP deve ser o primeiro elemento.
- Todos os filhos sob `soapenv:Header` devem possuir espaço de nomes qualificado.

- Um soapenv:Header pode ter estes atributos: actor, mustUnderstand e estilo de codificação.

A hierarquia de apelidos para acessar os cabeçalhos SOAP requer que o apelido raiz aponte para o envelope SOAP. Os elementos do cabeçalho e do corpo são apelidos-filhos do apelido raiz. É possível criar os apelidos emitindo a instrução CRIAR APELIDO. A seguir, você pode criar consultas que acessam informações que estão armazenadas no cabeçalho SOAP.

Exemplo

Estas instruções CREATE NICKNAME ilustram como criar apelidos de raiz, corpo e cabeçalho.

```
CREATE NICKNAME
STOCKSERVICE_GETLASTSELLPRICE_ROOT_NN
  (NN_PKEY VARCHAR(16)NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER "WSSERVER_HDR"
OPÇÕES
  (URL 'http://wswrap.svl.ibm.com:9081/StockPrice/services/StockService' ,
  SOAPACTION ' ' , TEMPLATE '<soapenv:Envelope>&STOCKSERVICE_GETLASTSELLPRICE_HEADER_NN[1,1]
  &STOCKSERVICE_GETLASTSELLPRICE_BODY_NN[1,1]</soapenv:Envelope>', XPATH '/soapenv:Envelope',
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
  ns1="http://soapheader.ibm.com" ');

CREATE NICKNAME STOCKSERVICE_GETLASTSELLPRICE_HEADER_NN
  (NN_PKEY VARCHAR(16)NOT NULL OPTIONS(PRIMARY_KEY 'YES'),
  NN_FKEY VARCHAR (16) NOT NULL OPTIONS(FOREIGN_KEY
'STOCKSERVICE_GETLASTSELLPRICE_ROOT_NN'),
  QUOTE_TIMESTAMP VARCHAR(30)OPTIONS(TEMPLATE '&column'))
FOR SERVER "WSSERVER_HDR"
OPÇÕES
  (XPATH './soapenv:Header',TEMPLATE '<soapenv:Header><ns1:quote_timestamp
>&quote_timestamp[1,1]
</ns1:quote_timestamp></soapenv:Header>',
  NAMESPACES 'ns1="http://soapheader.ibm.com",soapenv=
"http://schemas.xmlsoap.org/soap/envelope/" ');

CREATE NICKNAME STOCKSERVICE_GETLASTSELLPRICE_BODY_NN
  (NN_PKEY VARCHAR(16)NOT NULL
  OPTIONS(PRIMARY_KEY 'YES'), NN_FKEY VARCHAR 16)NOT NULL
  OPTIONS(FOREIGN_KEY 'STOCKSERVICE_GETLASTSELLPRICE_ROOT_NN'),
  GETLASTSELLPRICE_TICKER VARCHAR (48)
  OPTIONS(TEMPLATE '&column'), GETLASTSELLPRICERETURN VARCHAR (48
  OPTIONS(XPATH './ns1:getLastSellPriceResponse/ns1:getLastSellPriceReturn/text()'))
FOR SERVER "WSSERVER_HDR"
OPÇÕES
  (TEMPLATE '<soapenv:Body><ns1:getLastSellPrice><ns1:ticker>&getLastSellPrice_ticker[1,1]
</ns1:ticker></ns1:getLastSellPrice></soapenv:Body>', XPATH './soapenv:Body',
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
  ns1="http://soapheader.ibm.com" ');
```

A consulta a seguir retorna o preço da das ações IBM para uma data e hora específicas:

```
SELECT * from
  STOCKSERVICE_GETLASTSELLPRICE_ROOT_NN as a,
  STOCKSERVICE_GETLASTSELLPRICE_HEADER_NN as b,
  STOCKSERVICE_GETLASTSELLPRICE_BODY_NN as c
where
  a.nn_pkey = b.nn_fkey and
  a.nn_pkey = c.nn_fkey and
  b.QUOTE_TIMESTAMP='2007-05-09T18:30:20.000Z'and
  c.GETLASTSELLPRICE_TICKER='IBM';
```

Apelidos e Expressões XPath

O wrapper de serviços da Web utiliza apelidos que correspondem à estrutura em árvore dos dados XML. Cada apelido é definido por expressões XPath que representam valores de saída.

Os apelidos correspondem à estrutura em árvore dos dados de seus documentos XML. Os apelidos pais e filhos correspondem à estrutura raiz e aos elementos aninhados da estrutura em árvore de dados. Esses apelidos pais e filhos são conectados por chaves primárias e estrangeiras, especificadas com a instrução CREATE NICKNAME.

Cada apelido é definido por expressões XPath que representam valores de saída. O wrapper de serviços da Web utiliza expressões XPath para estabelecer uma correspondência entre os dados de um documento XML e as linhas de uma tabela relacional. Essas expressões XPath identificam os valores no documento XML e determinam como esses valores correspondem às colunas de cada linha. O wrapper apenas lê os dados XML e não atualiza o documento. A opção XPath contém as informações para localizar mensagens SOAP através do envelope SOAP e marcações de corpo SOAP. A mensagem getQuote está contida nos elementos de corpo e envelope SOAP.

A expressão XPath para a opção NICKNAME indica tags de repetição que estão no elemento de saída. A expressão XPath determina quantas ou quais linhas estarão no apelido. A expressão XPath da opção de coluna é relativa à expressão XPath NICKNAME. A expressão XPath da opção de coluna identifica os valores em uma linha. Uma expressão XPath da opção NICKNAME em um apelido filho é relativa a uma expressão XPath da opção NICKNAME em um apelido pai.

Ao criar um apelido, são escolhidas opções em um documento de linguagem de descrição de serviços da Web (WSDL) que especifica a associação entre o apelido e o documento XML.

Opção TEMPLATE no Apelido e nos Níveis de Coluna

Quando um apelido é criado, o wrapper de serviços da Web requer fragmentos de modelo em nível de apelido e nível de coluna, que é a opção TEMPLATE na instrução CREATE NICKNAME.

O wrapper de serviços da Web constrói documentos XML necessários ao ambiente de serviços da Web. O wrapper utiliza essas informações durante o planejamento da consulta e nas fases de execução da consulta.

A opção TEMPLATE para o Wrapper de Serviços da Web:

O wrapper de serviços da Web precisa de uma opção TEMPLATE na instrução CREATE NICKNAME no momento em que o apelido é criado.

Wrapper de Serviços da Web

Para o wrapper de serviços da Web, os atributos requeridos e opcionais variam de acordo com as definições no documento WSDL e de como a coluna é derivada. Uma coluna pode ser derivada de um elemento ou de um atributo de um elemento.

- Se a coluna é derivada de um elemento, então o valor minOccurs determina se uma coluna é opcional.
- Se o valor de minOccurs for igual a 0, então a coluna é opcional.

- Se o valor de minOccurs for igual a 1, então a coluna é requerida.
- Se a coluna é derivada de um atributo de um elemento, então o valor use no atributo determina se uma coluna é opcional.
- Se um atributo contém o valor use=optional, então a coluna é opcional.
- Se um atributo contém o valor use=required, então a coluna é requerida.

O exemplo a seguir é um atributo em uma definição de esquema que está associada com uma coluna:

```
<xsd:complexType>
  <xsd:sequence>
    <xsd:element ref="tns:ZooName"/>
    <xsd:element ref="tns:Count"/>
    <xsd:element ref="tns:LastModified"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:Zookeeper"/>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:string" use="optional"/>
</xsd:complexType>
```

Criando Visualizações Federadas para Apelidos de Serviços da Web

Crie visualizações federadas para apelidos de serviços da Web para garantir que as consultas que unem partes de uma hierarquia de apelidos de serviços da Web sejam executadas corretamente.

Sobre Esta Tarefa

Você pode definir visualizações federadas para a hierarquia de apelidos que descreve documento de serviços da Web. Uma visualização federada garante que as consultas que unem partes de uma hierarquia de apelidos de serviços da Web sejam executadas corretamente.

O wrapper de serviços da Web requer uniões de qualquer apelido-filho para incluir todos os nomes de volta no apelido-pai. Uma instrução SELECT em apenas um apelido-filho falha se o apelido-pai não estiver incluído na instrução. O método preferencial para inclusão de todos os apelidos necessários em uma consulta é definir uma visualização federada.

Procedimento

1. Defina uma visualização que inclua todos os apelidos que estão relacionados a uma operação no serviço da Web se você desejar unir tais apelidos.
2. Na cláusula WHERE da visualização, utilize os predicados de junção para todas as colunas que estão relacionadas através das opções de coluna PRIMARY_KEY e FOREIGN_KEY.

Exemplo

No exemplo a seguir, a chave primária está na coluna ooport_getzooreport_pk no apelido zooport_getzooreport_report_nn. A chave estrangeira está na coluna ooport_getzooreport_fkey no apelido zooport_getzooreport_report_report_nn.

```
CREATE VIEW zooreport
(zooId, zooName, number_of_zookeeper,
lastmodified, zookeeper_id, zookeeper_name,
cages, animal_name, animal_species, animal_lot)
AS ( SELECT zooId, report_zooName,
report_count, report_lastmodified,
zookeeper_id, zk.report_name, report_numbercages,
a.report_name, report_species,
```



```

report_lot
FROM zooport_getzooreport_nn ,
     zooport_getzooreport_report_nn as zk,
     zooport_getzooreport_report_report_nn as a
WHERE zk.ooport_getzooreport_pkey=a.ooport_getzooreport_fkey
AND zooport_getzooreport_pkey=zk.ooport_getzooreport_fkey);

```

Você pode obter informações de todos os apelidos com a seguinte instrução SELECT:

```
SELECT * FROM zooreport WHERE zooid='1';
```

Restrições de consulta para Wrappers de Serviços da Web

O uso de predicados e junções de consulta está restrito a serviços da Web conforme mostrado nos exemplos fornecidos.

Predicados Iguais

Os únicos predicados válidos nas colunas de entrada são os predicados iguais. Para colunas de saída, qualquer predicado é válido.

O exemplo a seguir retorna um erro com uma mensagem indicando que o predicado não é suportado nessa coluna. Nesse exemplo, a coluna zipcode é uma coluna de entrada:

```
SELECT return FROM gettemp WHERE zipcode<'95141'
```

O exemplo a seguir mostra uma consulta válida que utiliza um predicado igual nas colunas de entrada. O apelido customers é associado a uma tabela local que contém os IDs dos clientes. A consulta contém um predicado adicional na coluna Sales, que é uma coluna somente de saída.

```

SELECT a.name, a.address
FROM customers a, local_table b
WHERE
  a.customer_id=b.custid AND
  a.Sales > 300000;

```

Predicados para Colunas de Entrada Necessárias

Você deve fornecer valores de predicado de igualdade para todas as colunas de entrada necessárias em suas consulta SQL da hierarquia de apelidos à qual você faz referência. O wrapper retorna um SQLCODE -901 para todas as consultas que violam essa restrição.

Predicados IN ou OR

Para wrappers de serviços da Web, nenhum predicado IN ou OR é permitido para colunas de entrada.

Os exemplos a seguir mostram consultas inválidas. O apelido customers tem uma coluna de entrada necessária, customer_id:

```

SELECT * FROM customers
WHERE customer_id IN (12345, 67890, 11223);
SELECT * FROM customers
WHERE customer_id IN (SELECT custid FROM local_table); )

```

Junções em Colunas de Entrada Opcionais

Os exemplos a seguir demonstram uma restrição sobre a junção de colunas de entrada opcionais e o uso de variáveis de host. Você não pode juntar as colunas de entrada opcionais de uma tabela ou um apelido local. Se o WSDL gerar uma coluna de apelido de entrada como opcional e você precisar utilizar essa coluna em uma junção, deverá editar a instrução DDL para alterar a coluna para uma coluna de entrada obrigatória.

Neste exemplo, um apelido de wrapper de serviço da Web denominado order é criado com shipping_method como uma coluna de entrada opcional. A instrução a seguir é uma consulta válida, pois ela utilizar um literal no predicado:

```
SELECT * FROM order
  WHERE part="hammer" AND shipping_method="FEDEX";
```

Entretanto, se for incluída uma tabela local denominada orderparts, que define peças e métodos de remessa, na consulta, e a tabela tiver uma coluna denominada shipping_method que é opcional, a instrução é inválida:

```
SELECT * FROM
  order o, orderparts op
  WHERE
    o.part="hammer" AND
    o.shipping_method=op.shipping_method
```

Se forem utilizadas variáveis de host em consultas com predicados em colunas de entrada opcionais, a instrução também será inválida:

```
SELECT * FROM
  order o
  WHERE
    o.part=:hammer AND
    o.shipping_method=:shipping_method
```

Para garantir resultados válidos, as colunas de entrada associadas devem ser colunas obrigatórias para os wrappers de Serviços da Web.

Junções Externas

As junções externas entre apelidos que utilizam a chave primária de um apelido pai e a chave estrangeira de colunas de apelidos filhos não são suportadas.

Quando um elemento-pai em um documento XML não contém nenhum elemento-filho e se você utilizar uma junção interna entre o apelido pai e o apelido filho, nenhuma linha será retornada para esse elemento. Por exemplo, para determinado cliente, se não houver informações de detalhes bancários no sistema SAP, nenhuma linha será retornada para o apelido sap_bapi_customer_getdetail2_sap_customerbankdetail_NN em relação ao cliente específico.

As seguintes instruções CREATE NICKNAME definem as colunas que são utilizadas na consulta de exemplo:

```

CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
...
  NAME VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
               ns1:sap_customeraddress/ns1:NAME/text()'),
...
  NN_PKEY VARCHAR(16) OPTIONS(PRIMARY_KEY 'YES'),
  COMPANYCODE VARCHAR(4) OPTIONS(XPATH './ns3:COMPANYCODE/text()',
    TEMPLATE '<ns3:COMPANYCODE>&column</ns3:COMPANYCODE>'),
  CUSTOMERNO VARCHAR(10) OPTIONS(XPATH './ns3:CUSTOMERNO/text()',
    TEMPLATE '<ns3:CUSTOMERNO>&column</ns3:CUSTOMERNO>'),
...
  FOR SERVER sap_server
  OPTIONS(XPATH '//ns3:sap_bapi_customer_getdetail2',
    TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
              &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
              &COMPANYCODE[0,1]
              &CUSTOMERNO[1,1]
              </ns3:sap_bapi_customer_getdetail2>',
...

```

Figura 22. Resumo do Apelido getdetail2

```

CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_NN(
  CUSTOMER VARCHAR(10) OPTIONS(XPATH './ns2:CUSTOMER/text()',
    TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
  BANK_KEY VARCHAR(15) OPTIONS(XPATH './ns2:BANK_KEY/text()',
    TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
  BANK_ACCT VARCHAR(18) OPTIONS(XPATH './ns2:BANK_ACCT/text()',
    TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
  CTRL_KEY VARCHAR(2) OPTIONS(XPATH './ns2:CTRL_KEY/text()',
    TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
  BANK_REF VARCHAR(20) OPTIONS(XPATH './ns2:BANK_REF/text()',
    TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),
  NN_FKEY VARCHAR(16) OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN'))
  FOR SERVER sap_server
  OPTIONS(XPATH './ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
    TEMPLATE '<ns3:sap_customerbankdetail>
              <ns2:sap_customerbankdetail>
                &CUSTOMER[0,1]
                &BANK_KEY[0,1]
                &BANK_ACCT[0,1]
                &CTRL_KEY[0,1]
                &BANK_REF[0,1]
              </ns2:sap_customerbankdetail>
              </ns3:sap_customerbankdetail>',
...

```

Figura 23. Resumo do Apelido de Detalhes Bancários do Cliente

No exemplo a seguir, a consulta não retorna linhas, pois há uma junção interna entre os dois apelidos:

```

SELECT a.name, b.bank_key
  FROM sap_bapi_customer_getdetail2_NN a,
       sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
 WHERE a.customerno='1234567890'
       AND a.NN_PKEY=b.NN_FKEY;

```

Se uma definição de apelido de wrapper dos serviços da Web contiver as colunas de entrada necessárias, então uma junção externa esquerda entre esse apelido e qualquer outra tabela local ou outros apelidos não será suportada.

Origens de Dados dos Serviços da Web - Exemplo de Consultas

Exemplos de consultas SQL para o wrapper de serviços da Web são mostradas.

Exemplo 1: Utilizando Tabelas de Consultas Materializadas

Você utiliza tabelas de consultas materializadas para armazenar dados em cache localmente e para aprimorar o desempenho de suas consultas. Você pode utilizar apelidos das origens de dados de serviços da Web para criar tabelas de consultas materializadas. Para algumas consultas, o banco de dados pode determinar automaticamente se a tabela de consulta materializada pode responder a uma consulta sem acessar as tabelas de base. O procedimento a seguir mostra como criar e preencher uma tabela de consulta materializada:

1. Crie uma tabela local ou de base:

```
CREATE TABLE mystocks(ticker VARCHAR(10));
```

Você pode utilizar a tabela local para manter todos os valores que deseja armazenar em cache.

2. Insira todos os valores que deseja armazenar em cache na tabela:

```
INSERT INTO mystocks VALUES('IBM');
INSERT INTO mystocks VALUES('MSFT');
...
```

3. Crie um apelido de serviços da Web:

```
CREATE NICKNAME stockquote_nn (
    ticker VARCHAR(40) OPTIONS (TEMPLATE '&column'),
    price VARCHAR(16) OPTIONS (XPATH './Result/text()')
)
FOR SERVER stock_server
OPTIONS (TEMPLATE '<ticker>&column</ticker>'
        XPATH './Result/text()' );
```

4. Crie uma visualização que consista no apelido e na tabela local:

```
CREATE VIEW stock_quote_view (ticker, price)
AS (
    SELECT nn.ticker, nn.price
    FROM stockquote_nn nn, mystocks s
    WHERE nn.ticker=s.ticker
);
```

5. Crie uma tabela de consulta materializada:

```
CREATE TABLE stockquote_MQT (ticker, ticker2, price)
as (SELECT nn.ticker,s.ticker as ticker2, nn.price
FROM stockquote_nn nn, mystocks s
WHERE nn.ticker=s.ticker )
DATA INITIALLY DEFERRED REFRESH DEFERRED;
```

Inclua todas as colunas VARCHAR no predicado de junção (nn.ticker e s.ticker) na lista de saída da tabela de consulta materializada para maximizar as oportunidades de que uma tabela de consulta materializada seja utilizada pelo banco de dados federado.

Para adiar a atualização da tabela de consulta materializada, especifique a palavra-chave REFRESH DEFERRED. As tabelas de consultas materializadas que são especificadas com a palavra-chave REFRESH DEFERRED não refletem

as alterações na tabela de base subjacente. Utilize a cláusula DATA INITIALLY DEFERRED para que seus dados não sejam inseridos na tabela como parte da instrução CREATE TABLE.

6. Emita uma instrução REFRESH TABLE para preencher a tabela. Os dados na tabela refletem o resultado da consulta como uma captura instantânea do momento em que você emite a instrução REFRESH TABLE. O exemplo a seguir preenche a tabela stockquote_MQT e configura um valor para o registro especial com o período de atualização atual.

```
REFRESH TABLE stockquote_MQT;
```

```
SET CURRENT REFRESH AGE any;
```

As consultas que são executadas nos dados na tabela de consulta materializada são mais rápidas do que as consultas executadas em uma tabela de base. Quando você deseja utilizar a tabela de consulta materializada, deve consultar a visualização e não o apelido:

```
SELECT * FROM stock_quote_view
WHERE ticker='IBM';
```

Se você emitir uma consulta para selecionar um valor que não foi armazenado em cache, 0 linhas são retornadas.

Exemplo 2: Emitindo Junções Utilizando as Chaves Primária e Estrangeira

As colunas PRIMARY_KEY e FOREIGN_KEY definem relacionamentos entre os apelidos-pai e seus filhos. Cada apelido-pai deve possuir uma coluna de opção de chave primária. Defina o filho de um apelido pai com a opção de coluna da chave externa que faz referência à coluna de chaves primárias de um apelido pai. Um apelido pode ter vários filhos, mas um apelido pode ter apenas um pai.

Como estas colunas contêm apenas dados binários, as colunas não definidas com as palavras-chave FOR BIT DATA NOT NULL. Você pode definir explicitamente as colunas PRIMARY_KEY e FOREIGN_KEY como FOR BIT DATA NOT NULL ao criar o apelido.

O exemplo a seguir mostra como o wrapper de serviços da Web utiliza as colunas PRIMARY_KEY e FOREIGN_KEY para associar apelidos-pai e seus filhos.

```
CREATE NICKNAME zooport_getzooreport_nn (
  zooid VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  zoo_id VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/@id'),
  report_zoo_zooname VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:ZooName/text()'),
  report_zoo_count VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:Count/text()'),
  report_zoo_lastmodified VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:LastModified/text()'),
  nn_pk VARCHAR (16) NOT NULL OPTIONS(PRIMARY_KEY 'YES'),
  url VARCHAR (256) OPTIONS(URLCOLUMN 'Y'),
  soapaction VARCHAR (256) OPTIONS(SOAPACTIONCOLUMN 'Y')
) FOR SERVER "mytestsrvr"
OPTIONS (
  URL 'http://localhost:9080/MaelstromTest/services/ZooPort',
  SOAPACTION 'http://myzoo.com/getZooReport' ,
  TEMPLATE '<soapenv:Envelope>
    <soapenv:Body>
      <zooId&zooId[1,1]</zooId>
    </soapenv:Body>
  </soapenv:Envelope>',
```

```

        XPATH '/soapenv:Envelope/soapenv:Body' ,
        NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
                    ns1="http://myzoo.com" ');
CREATE NICKNAME zooport_getzooreport_report_zookeeper_nn (
  nn_fk VARCHAR (16) NOT NULL
    OPTIONS(FOREIGN_KEY 'ZOOPORT_GETZOOREPORT_NN'),
  zookeeper_id VARCHAR (48) OPTIONS(XPATH './@id'),
  report_zookeeper_name VARCHAR (48) OPTIONS(XPATH './ns1:Name/text()'),
  zookeeper_numbercages VARCHAR(48)
    OPTIONS(XPATH './ns1:NumberOfCages/text()'),
  nn_pk VARCHAR (16) NOT NULL OPTIONS(PRIMARY_KEY 'YES')
)
FOR SERVER "MYTESTSRVR" OPTIONS(
  XPATH './ns1:Zoo/ns1:Zookeeper' ,
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
              ns1="http://myzoo.com" ');

```

A chave estrangeira, `nn_fk`, no apelido `zooport_getzooreport_report_zookeeper_nn`, faz referência ao apelido-pai, `zooport_getzooreport_nn` na opção de chave estrangeira. As colunas de apelido de chave primária e estrangeira designadas não correspondem aos dados em seu documento WSDL pois estas colunas de apelido contêm chaves que são geradas pelo wrapper. Estas chaves identificam um relacionamento entre os apelidos-pai e seus filhos, que é exclusivo apenas em uma consulta. Se o apelido-filho contém uma coluna de entrada, o modelo da opção de apelido-pai faz referência a este apelido-filho na estrutura do modelo com a opção de apelido.

A seguinte instrução SQL une os apelidos-pai e seus filhos:

```

SELECT *
FROM   zooport_getzooreport_nn a,
       zooport_getzooreport_report_zookeeper_nn z,

WHERE  a.nn_pk = z.nn_fk
       AND a.zooid = 100
;

```

A descrição a seguir explica como o wrapper de serviços da Web utiliza as opções de apelido e coluna `TEMPLATE` e `XPATH` durante a execução da consulta. Ela não representa um exemplo de uma implementação específica.

Quando você une as colunas de chave primária e estrangeira, o wrapper de serviços da Web envia uma mensagem para o fornecedor de serviços da Web e um conjunto de linhas é retornado do fornecedor de serviços da Web. O wrapper gera uma mensagem para o apelido-pai substituindo os valores da coluna de entrada (`a.zooid = 100`) na consulta para a referência no modelo de opção de coluna (`ZOOID VARCHAR (48) OPTIONS(TEMPLATE '&column')`), e então todas as referências de coluna na opção de modelo do apelido (`<zooId>&zooId[1,1]</zooId>`). A opção de modelo do apelido pode incluir referências a colunas ou referências a apelidos-filhos. A mensagem então é enviada para o serviço da Web.

O wrapper gera as linhas para um apelido aplicando a opção de apelido `XPATH` no documento retornado pelo serviço da Web. Se a opção de apelido `XPATH` retorna vários fragmentos XML, então o apelido contém várias linhas. O opção `XPATH` de coluna é aplicada nos fragmentos XML resultantes que representam as linhas para obter os valores da coluna. Se um apelido possui um ou mais pais indiretos, todas as expressões `XPATH` do apelido-pai são aplicadas do início da hierarquia para baixo, antes que a opção de apelido `XPATH` e a opção de coluna `XPATH` sejam aplicadas para este apelido.

Configurando o Acesso às Origens de Dados do XML

Você pode integrar os dados que estão nas origens de dados XML com informações de outras origens utilizando um sistema federado.

Sobre Esta Tarefa

Para configurar um servidor federado para acessar as origens de dados XML, você deve fornecer ao servidor federado informações sobre as origens de dados e objetos que deseja acessar. Após configurar o servidor federado, você pode criar consultas para acessar as origens de dados XML.

wrapper do XML

O XML (Linguagem de Marcação Extensível) é um formato universal para documentos e dados estruturados. O XML utiliza tags para estruturação dos dados em documentos.

Os arquivos XML utilizam a extensão de arquivo .xml. Como o HTML, o XML utiliza palavras que são colocadas entre parênteses (< >) como tags. As tags estruturam os dados que estão no documento.

O documento a seguir é um documento XML de amostra.

```
<?xml version="1.0" encoding="UTF-8"?>
<doc>
  <customer id='123'>
    <name>...</name>
    <address>...</address>
    ...
    <order>
      <amount>...</amount>
      <date>...</date>
      <item quant='12'>
        <name>...</name>
      </item>
      <item quant='4'>...</item>
      ...
    </order>
    <order>...</order>
    ...
    <payment>
      <number>...</number>
      <date>...</date>
    </payment>
    <payment>>...</payment>
    ...
  </customer>
  <customer id='124'>...</customer>
</doc>
```

Figura 24. Documento XML de Amostra

Como Funciona o Wrapper XML

O wrapper XML permite utilizar o SQL para consultar os seguintes tipos de dados:

- Documentos XML externos que estão armazenados em um único arquivo
- Vários arquivos XML em um caminho de diretório

- Arquivos XML remotos que são referidos com uma URI (Uniform Reference Identifier)
- Documentos XML armazenados em colunas relacionais

A figura a seguir mostra como o wrapper XML funciona com seu sistema federado.

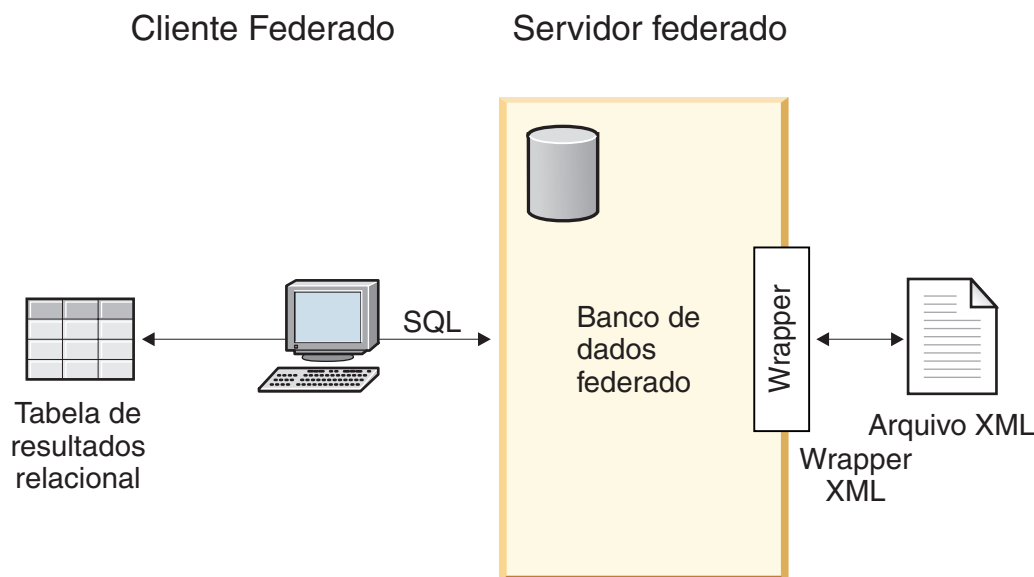


Figura 25. Como Funciona o Wrapper XML

Com o wrapper XML, você pode mapear dados XML de uma origem de dados externa para um esquema relacional composto de um conjunto de apelidos. A estrutura de um documento XML é logicamente equivalente a um esquema relacional no qual os elementos aninhados e repetitivos são modelados como tabelas separadas com chaves externas.

Os apelidos que correspondem a um documento XML são organizados em uma estrutura em árvore na qual os apelidos-filho mapeiam para elementos que aninhados dentro do elemento que corresponde ao apelido-pai.

Um apelido raiz é um apelido no nível superior de uma hierarquia de apelidos. Um apelido não-raiz é um apelido que possui um apelido-pai em uma hierarquia de apelidos. Você pode ter apelidos raiz que não são o elemento no nível superior em um documento XML.

Quando os elementos aninhados são repetidos ou possuem identidades distintas com estruturas complexas, você pode fornecer apelidos separados para cada elemento aninhado.

Os apelidos filho e pai são conectados pelas chaves primárias e externas geradas pelo wrapper.

As expressões XPath são utilizadas para mapear um documento XML para um esquema relacional composto de um conjunto de apelidos. O XPath é um mecanismo de endereçamento para identificação de partes de um arquivo XML (por exemplo, os grupos de nós e atributos dentro de uma árvore de documentos XML). A sintaxe XPath básica é semelhante ao endereçamento do sistema de arquivos.

Cada apelido é definido por uma expressão XPath que identifica os elementos XML representando tuplas individuais, e um conjunto de expressões XPath que identifica como extrair os valores de colunas de cada elemento.

Um exemplo de mapeamento de documentos XML

O exemplo a seguir ilustra como:

- O documento XML de amostra é mapeado para um conjunto de apelidos
- Relacionamentos pai e filho são estabelecidos utilizando chaves primárias e externas
- Expressão XPath são utilizadas para definir tuplas e colunas individuais e dentro de cada elemento do documento
- Uma consulta pode ser executada no documento XML após o documento ser registrado para seu sistema federado

O documento XML de amostra contém um conjunto de elementos de clientes. Cada elemento engloba diversos elementos de pedido e pagamento.

Os elementos de pedido englobam diversos elementos de itens.

O relacionamento entre os elementos é mostrado na figura a seguir.

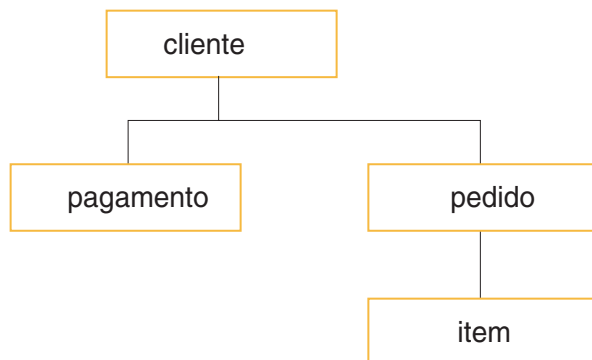


Figura 26. Estrutura em Árvore do Documento XML de Amostra

Nesta estrutura, você pode utilizar a instrução CREATE NICKNAME para mapear o documento XML para um esquema relacional que inclui quatro apelidos:

- customers
- orders
- payments
- items

Defina relacionamentos entre os apelidos especificando cada apelido como um apelido pai ou um apelido filho utilizando opções de colunas de apelidos especiais que especificam chaves primárias e externas. Cada apelido pai deve ter uma coluna especial que é atribuída com uma opção de coluna de chave primária. Defina o filho de um apelido pai com a opção de coluna da chave externa que faz referência à coluna de chaves primárias de um apelido pai. As colunas de apelidos primários e externos atribuída não correspondem aos dados em seu documento XML porque estas colunas de apelidos conterão chaves que são geradas pelo wrapper. Um apelido pode ter vários filhos, mas um apelido pode ter apenas um pai. O apelido raiz não possui nenhum pai.

Para o documento XML de amostra, o apelido `customers` possui uma chave primária definida, e os apelidos `orders`, `payments` e `items` possuem chaves externas definidas que apontam para o apelido pai. As chaves externas dos apelidos `orders` e `payments` apontam para o apelido `customers`, e a chave externa do apelido `items` aponta para o apelido `orders`.

Para identificar os elementos XML que representam tuplas individuais, crie uma expressão XPath. Neste exemplo, todos os elementos de clientes são referidos utilizando a expressão XPath `'/doc/customer'`, e todos os elementos de pedidos são referidos utilizando a expressão XPath `'./order'`. O ponto na expressão XPath `'./order'` indica que as tuplas de cada elemento de pedido estão aninhadas dentro de tuplas dos elemento de cliente correspondente.

Crie um conjunto de expressões XPath para especificar como extrair os valores de colunas de cada elemento. Neste exemplo, o atributo `id` dos elementos de clientes, agora uma coluna definida no apelido, são referidos utilizando a expressão XPath `'./@id'`. O elemento de nomes dos elementos de clientes é referido utilizando a expressão XPath `'./name'`. O elemento de endereço dos elementos de clientes é referido utilizando a expressão XPath `'./address/@street'`.

Após mapear o documento XML em um conjunto de apelidos utilizando a instrução `CREATE NICKNAME`, defina cada apelido como um pai ou um filho utilizando chaves primárias e externas. Especifique expressões XPath nestas chaves primárias e externas que definem tuplas e colunas individuais dentro de cada elemento do documento. Então, poderá executar consultas SQL no documento XML.

Incluindo XML em um Sistema Federado

Para configurar um servidor federado para acessar as origens de dados XML, você deve fornecer ao servidor federado informações sobre as origens de dados e objetos que deseja acessar.

Antes de Iniciar

- A federação deve ser instalada em um servidor que atuará como o servidor federado.
- Um banco de dados deve existir no servidor federado.

Procedimento

1. Registrar o wrapper XML.
2. Registrar a definição do servidor XML.
3. Registrar apelidos para as origens de dados XML.
4. Criar visualizações federadas para apelidos não-raiz.

Registrando o Wrapper XML

Você deve registrar um wrapper para acessar origens de dados XML.

Sobre Esta Tarefa

Servidores federados utilizam wrappers para se comunicar com e recuperar dados de origens de dados. Wrappers são implementados como um conjunto de arquivos de biblioteca.

Se você utilizar um servidor de proxy e um keystore para acessar arquivos XML, você pode especificar as informações do servidor de proxy como opções ao

registrar a definição do wrapper ou do servidor. Se você especificar as informações do servidor de proxy ao registrar a definição do servidor, essas configurações anulam as configurações do servidor de proxy especificadas ao registrar o wrapper.

Procedimento

Emita a instrução CRIAR WRAPPER a partir da linha de comandos.

```
CREATE WRAPPER wrapper_name
LIBRARY library_name;
```

Se você utilizar um servidor de proxy ou um keystore para acessar arquivos XML, você deve especificar diversas opções ao registrar o wrapper XML ou a definição do servidor. Para especificar as informações do servidor de proxy ao registrar o wrapper XML, a instrução emitida é a seguinte:

```
CREATE WRAPPER wrapper_name
LIBRARY library_name
OPTIONS (PROXY_TYPE 'type',
PROXY_SERVER_NAME 'server_name',
PROXY_SERVER_PORT 'port_number');
```

Especifique o parâmetro LIBRARY na instrução CREATE WRAPPER. O nome do arquivo de biblioteca do wrapper especificado depende do sistema operacional do servidor federado. Consulte a lista de arquivos da biblioteca do wrapper XML para obter o nome da biblioteca correta para especificação na instrução CREATE WRAPPER.

Se você utilizar um servidor de proxy ou um keystore para acessar arquivos XML, você deve especificar diversas opções ao registrar o wrapper XML ou a definição do servidor.

Arquivos da Biblioteca do Wrapper XML:

Os arquivos da biblioteca do wrapper XML são incluídos no servidor federado ao instalar o servidor federado.

Ao instalar o IBM InfoSphere Federation Server, três arquivos de bibliotecas são incluídos no caminho do diretório padrão. Por exemplo se o servidor federado estiver em execução no AIX, os arquivos da biblioteca do wrapper que são incluídos no caminho do diretório são libdb2lsxml.a, libdb2lsxmlF.a e libdb2lsxmlU.a. O arquivo da biblioteca do wrapper padrão é libdb2lsxml.a. Os outros arquivos da biblioteca do wrapper são utilizados com opções específicas do wrapper.

Inclua o parâmetro LIBRARY na instrução CREATE WRAPPER e especifique o nome do arquivo da biblioteca padrão do wrapper.

Os caminhos padrão do diretório e os nomes de arquivos de biblioteca padrão do wrapper estão listados na tabela a seguir.

Tabela 40. Locais e Nomes de Arquivos da Biblioteca do Wrapper XML

Sistema Operacional	Caminho do Diretório	Nome do arquivo da biblioteca do wrapper
AIX	/usr/opt/<install_path>/lib32/ /usr/opt/<install_path>/lib64/	libdb2lsxml.a
Linux	/opt/IBM/db2/<install_path>/lib32 /opt/IBM/db2/<install_path>/lib64	libdb2lsxml.so

Tabela 40. Locais e Nomes de Arquivos da Biblioteca do Wrapper XML (continuação)

Sistema Operacional	Caminho do Diretório	Nome do arquivo da biblioteca do wrapper
Solaris	/opt/IBM/db2/<install_path>/lib32 /opt/IBM/db2/<install_path>/lib64	libdb2lsxml.so
Windows	%DB2PATH%\bin	db2lsxml.dll

<install_path> é o caminho do diretório em que o servidor federado está instalado no Linux ou UNIX.

%DB2PATH% é a variável de ambiente utilizada para especificar o caminho do diretório onde o servidor federado está instalado no Windows. O caminho padrão do caminho do diretório do Windows é C:\Arquivos de Programa\IBM\SQLLIB.

Instrução CREATE WRAPPER - Exemplos para o Wrapper XML:

Utilize a instrução CREATE WRAPPER para registrar o wrapper XML. Os exemplos mostram os parâmetros necessários para acessar documentos XML com ou sem um servidor de proxy.

Registrando um Wrapper

Se não estiver utilizando um servidor de proxy para acessar documentos XML, a instrução emitida para registrar o wrapper é a seguinte:

```
CREATE WRAPPER xml_wrapper LIBRARY 'libdb2lsxml.a';
```

xml_wrapper

O nome que você atribui ao wrapper XML. Nomes de wrapper duplicados não são permitidos.

LIBRARY '*libdb2lsxml.a*'

O nome do arquivo de biblioteca para servidores federados que usam sistemas operacionais AIX.

Registrando um Wrapper para um Servidor Proxy HTTP

Para registrar um wrapper e especificar um servidor proxy HTTP, utilize a seguinte instrução:

```
CREATE WRAPPER xml_wrapper LIBRARY 'libdb2lsxml.a'
  OPTIONS (PROXY_TYPE 'HTTP', PROXY_SERVER_NAME 'proxy_http',
  PROXY_SERVER_PORT '8080');
```

PROXY_TYPE '*HTTP*'

Especifica o tipo de proxy que é utilizado para acessar a Internet quando se está por trás de um firewall. Os valores válidos são 'NONE', 'HTTP' ou 'SOCKS'.

PROXY_SERVER_NAME '*proxy_http*'

Especifica o nome do servidor proxy ou o endereço IP. Esta opção será necessária se o valor para a opção do servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

PROXY_SERVER_PORT '*8080*'

Especifica o número da porta do servidor proxy. Esta opção é necessária se o valor para a opção de servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

Registrando um Wrapper para um Servidor Proxy SOCKS

Para registrar um wrapper e especificar um servidor proxy SOCKS sem informações sobre autenticação, use a seguinte instrução:

```
CREATE WRAPPER xml_wrapper LIBRARY 'libdb2lsxml.so'  
  OPTIONS (PROXY_TYPE 'SOCKS', PROXY_SERVER_NAME 'proxy_socks',  
          PROXY_SERVER_PORT '1081');
```

LIBRARY '*libdb2lsxml.so*'

Especifica o nome do arquivo da biblioteca do wrapper para servidores federados que utilizam sistemas operacionais Linux e Solaris.

PROXY_TYPE '*SOCKS*'

Especifica o tipo de proxy que é utilizado para acessar a Internet quando se está por trás de um firewall. Os valores válidos são 'NONE', 'HTTP' ou 'SOCKS'.

PROXY_SERVER_NAME '*proxy_socks*'

Especifica o nome do servidor proxy ou o endereço IP. Esta opção será necessária se o valor para a opção do servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

PROXY_SERVER_PORT '*1081*'

Especifica o número da porta do servidor proxy. Esta opção é necessária se o valor para a opção de servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

Registrando a Definição do Servidor para uma Origem de Dados XML

Você deve registrar uma definição de servidor para recuperar documentos XML. A definição de servidor que você registra depende se você utiliza um servidor de proxy para acessar os documentos XML.

Sobre Esta Tarefa

Para recuperar documentos XML que estão atrás de um firewall utilizando um proxy e uma URI (Uniform Resource Identifier), você deve incluir as opções do servidor de proxy na instrução CREATE SERVER.

Se não estiver utilizando um servidor de proxy, você ainda deve registrar uma definição de servidor porque a hierarquia de objetos federados necessita que os arquivos XML sejam associados a um objeto de definição do servidor específico.

Procedimento

Emita a instrução CRIAR SERVIDOR a partir da linha de comandos.

Se você não utilizar um servidor de proxy para acessar documentos XML, a instrução emitida é a seguinte:

```
CREATE SERVER server_definition_name  
  WRAPPER wrapper_name;
```

Se você utilizar um servidor de proxy para acessar documentos XML, a instrução emitida é a seguinte:

```
CREATE SERVER server_definition_name  
  WRAPPER wrapper_name  
  OPTIONS (PROXY_TYPE 'type',  
          PROXY_SERVER_NAME 'server_name',  
          PROXY_SERVER_PORT 'port_number');
```

Se você utilizar um servidor de proxy para acessar documentos XML, o protocolo utilizado pode necessitar que você especifique um ID de autorização e uma senha para o servidor de proxy. Especifique as informações de autenticação como opções quando você registrar a definição do servidor.

Instrução CREATE SERVER - Exemplos para o Wrapper XML:

Utilize a instrução CREATE SERVER para registrar definições de servidores para o wrapper XML. Os exemplos mostram os parâmetros necessários para acessar documentos XML com ou sem um servidor de proxy.

Registrando uma Definição de Servidor

Mesmo se não estiver utilizando um servidor de proxy para acessar documentos XML, ainda será necessário registrar uma definição de servidor. A hierarquia de objetos federados necessita que os arquivos XML estejam associados a um objeto de definição de servidor específico. A instrução emitida para registrar uma definição de servidor é a seguinte:

```
CREATE SERVER xml_server WRAPPER xml_wrapper;
```

xml_server

Um nome que você atribui à definição do servidor XML. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER *xml_wrapper*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

Definições de servidor quando um servidor proxy é usado

Use as opções do servidor proxy na instrução CREATE SERVER se todas estas condições forem verdadeiras:

- Você deseja recuperar dados utilizando uma URI
- A URI utilizada recuperará dados atrás de um firewall, por meio de um proxy
- O firewall ou proxy utilizado é HTTP ou SOCKS

As opções que você especifica dependem do tipo do servidor de proxy que você deseja acessar.

Verifique com o administrador de rede as informações sobre o tipo de proxy que você utiliza, e as configurações que você deve especificar nas opções do proxy.

Registrando uma definição de servidor para um servidor de proxy HTTP

Para registrar uma definição de servidor e especificar um servidor de proxy HTTP, utilize a seguinte instrução:

```
CREATE SERVER xml_server_http  
  WRAPPER xml_wrapper  
  OPTIONS (PROXY_TYPE 'HTTP', PROXY_SERVER_NAME 'proxy_http',  
          PROXY_SERVER_PORT '8080');
```

xml_server_http

Um nome que você atribui à definição do servidor XML. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER *xml_wrapper*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

PROXY_TYPE 'HTTP'

Especifica o tipo de proxy que é utilizado para acessar a Internet quando se está por trás de um firewall. Os valores válidos são 'NONE', 'HTTP' ou 'SOCKS'.

PROXY_SERVER_NAME 'proxy_http'

Especifica o nome do servidor proxy ou o endereço IP. Esta opção será necessária se o valor para a opção do servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

PROXY_SERVER_PORT '8080'

Especifica o número da porta do servidor proxy. Esta opção é necessária se o valor para a opção de servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

Registrando uma definição de servidor para um servidor proxy SOCKS

Para registrar uma definição de servidor e especificar um servidor proxy SOCKS quando informações sobre autenticação não são necessárias, use esta instrução:

```
CREATE SERVER xml_server_socks
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS', PROXY_SERVER_NAME 'proxy_socks',
    PROXY_SERVER_PORT '1081');
```

xml_server_socks

Um nome que você atribui à definição do servidor XML. Nomes de definição de servidor duplicados não são permitidos.

WRAPPER *xml_wrapper*

O Nome do Wrapper Especificado na Instrução CREATE WRAPPER.

PROXY_TYPE 'SOCKS'

Especifica o tipo de proxy que é utilizado para acessar a Internet quando se está por trás de um firewall. Os valores válidos são 'NONE', 'HTTP' ou 'SOCKS'.

PROXY_SERVER_NAME 'proxy_socks'

Especifica o nome do servidor proxy ou o endereço IP. Esta opção será necessária se o valor para a opção do servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

PROXY_SERVER_PORT '1081'

Especifica o número da porta do servidor proxy. Esta opção é necessária se o valor para a opção de servidor PROXY_TYPE for 'HTTP' ou 'SOCKS'.

Registrando uma definição de servidor para um servidor proxy SOCKS com informações sobre autenticação

Para registrar uma definição de servidor e especificar um servidor proxy SOCKS sem informações sobre autenticação, use a seguinte instrução:

```
CREATE SERVER xml_server_socks
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS', PROXY_SERVER_NAME 'proxy_socks',
    PROXY_SERVER_PORT '1081', PROXY_AUTHID 'Martin',
    PROXY_PASSWORD 'not4me');
```

PROXY_AUTHID 'Martin'

Especifica o nome de usuário no servidor de proxy. Esta opção do servidor será necessária se o valor da opção de servidor PROXY_TYPE for 'SOCKS'.

PROXY_PASSWORD 'not4me'

Especifica a senha no servidor de proxy associada ao nome de usuário 'Martin'. Esta opção do servidor será necessária se o valor da opção de servidor PROXY_TYPE for 'SOCKS'.

Especificando um limite de tempo nas respostas do servidor de proxy

Além das opções de servidores necessárias para servidores de proxy, você também pode especificar a opção de servidor SOCKET_TIMEOUT ao registrar uma definição de servidor. A opção de servidor SOCKET_TIMEOUT especifica o quantidade máxima de tempo, em minutos, que o servidor federado aguardará pelos resultados do servidor de proxy. Se você não especificar a opção de servidor SOCKET_TIMEOUT, não haverá nenhum limite de tempo. O servidor federado aguardará indefinidamente pelos resultados do servidor de proxy.

Para registrar uma definição de servidor e especificar por quanto tempo o servidor federado aguardará por uma resposta do servidor de proxy, utilize a seguinte instrução:

```
CREATE SERVER xml_server_http
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'HTTP', PROXY_SERVER_NAME 'proxy_http',
    PROXY_SERVER_PORT '8080', SOCKET_TIMEOUT '12');
```

SOCKET_TIMEOUT 12

Especifica que o servidor federado aguardará por 12 minutos por uma resposta do servidor de proxy.

Acessar Arquivos XML Utilizando um Servidor Proxy

Se sua rede utiliza um servidor de proxy, você deve especificar informações sobre o servidor de proxy ao registrar a definição de wrapper ou de servidor para origens de dados XML.

As opções que você especifica dependem do tipo de servidor de proxy que deseja acessar, e se você está utilizando protocolos SSL (Secure Socket Layer) ou TLS (Transport Layer Security).

Você pode especificar as opções de proxy e SSL ao registrar um wrapper ou uma definição de servidor:

- Se você especificar essas opções ao registrar o wrapper, os apelidos que estão associados com esse wrapper utilizarão as opções que estão configuradas para o wrapper.
- Se você especificar essas opções ao registrar uma definição de servidor, os apelidos que estão associados a essa definição de servidor utilizarão as opções que estão configuradas para a definição de servidor.
- Se você especificar valores diferentes para essas opções ao registrar o wrapper e a definição de servidor, os valores que estão configurado para a definição de servidor têm prioridade sobre os valores que estão configurado para o wrapper.

O wrapper XML possui um recurso de validação que pode ter limitações quando utilizado com o servidor de proxy. As condições nas quais você observará esta limitação são:

- Você está utilizando o recurso de proxy, no nível do servidor, e você configurou as diversas opções de proxy
- O documento da instância XML contém uma referência a um esquema XML externo localizado fora do firewall

Se você possui uma destas condições, tente alterar o local do seu esquema XML para um local dentro do firewall. Se você alterar o local do esquema XML, será necessário atualizar o documento da instância XML com o novo local do esquema XML.

Opções do Servidor Proxy

A tabela a seguir lista as opções que devem ser especificadas para cada tipo de servidor proxy:

Tabela 41. Opções Que São Necessárias Com Servidores Proxy

Tipo de servidor proxy	Opções do wrapper ou do servidor necessárias
HTTP ou SOCKS sem autenticação	PROXY_TYPE PROXY_SERVER_NAME PROXY_SERVER_PORT
HTTP ou SOCKS com autenticação	PROXY_TYPE PROXY_SERVER_NAME PROXY_SERVER_PORT PROXY_AUTHID PROXY_PASSWORD

Opções do Servidor SSL

A tabela a seguir lista as opções que você deve especificar ao utilizar os protocolos SSL:

Tabela 42. Opções que são necessárias com protocolos SSL

Objeto federado	Opções necessárias
Wrapper	SSL_KEYSTORE_FILE SSL_KEYSTORE_PASSWORD SSL_VERIFY_SERVER_CERTIFICATE
Definição de servidor	SSL_KEYSTORE_FILE SSL_KEYSTORE_PASSWORD SSL_VERIFY_SERVER_CERTIFICATE SSL_CLIENT_CERTIFICATE_LABEL

Apelidos para Origens de Dados XML

Você deve registrar um apelido para cada documento XML que deseja acessar. Utilize estes apelidos, em vez dos nomes de documentos, ao consultar origens de dados XML.

Antes de registrar os apelidos, você deve compreender o seguinte:

Associações de Dados entre Apelidos e Documentos XML:

Os apelidos correspondem à estrutura em árvore dos dados de seus documentos XML. Os apelidos pais e filhos correspondem à estrutura raiz e aos elementos aninhados da estrutura em árvore de dados. Esses apelidos pais e filhos são conectados por chaves primárias e estrangeiras, especificadas com a instrução CREATE NICKNAME.

Cada apelido é definido pelas expressões XPath que desempenham as seguintes funções:

- Identifica os elementos XML que representam tuplas individuais
- Especifica como extrair os valores de colunas de cada elemento

O wrapper XML utiliza expressões XPath para estabelecer uma correspondência entre os dados no documento XML e as linhas em uma tabela relacional. Estas expressões XPath identificam os valores dentro do documento XML e determinam como estes valores correspondem às colunas de cada linha. O wrapper XML lê apenas os dados do documento XML. O wrapper XML não atualiza estes dados.

Ao criar um apelido, você escolhe as opções que especificam a associação entre o apelido e o documento XML. Os apelidos são associados aos seus documentos XML de forma fixa ou com nomes de origens que você especificar.

Com uma associação fixa, o apelido representa dados de documentos XML específicos. Estes documentos XML incluem:

Um arquivo local

Você especifica um arquivo XML como seu documento XML.

Vários arquivos locais em um caminho de diretório

Especifique um caminho de diretório no qual residem vários arquivos XML. Os arquivos XML neste caminho de diretório fornecem os dados de documentos XML para o apelido. Todos os arquivos XML devem ter a mesma configuração. Se algum arquivo XML no diretório possui uma configuração diferente da configuração do apelido, o wrapper XML retorna valores nulos ao processar esse arquivo de dados XML. O diretório deve estar local no servidor federado ou acessível em um sistema de arquivos compartilhado.

Ao varrer o diretório, o wrapper XML retém e analisa apenas os arquivos com uma extensão `.xml`. O wrapper XML ignora todos os outros arquivos, incluindo arquivos com uma extensão `.txt`, arquivos com uma extensão `.xsd` e arquivos sem extensões.

Utilize a opção `FILE_PATH` na instrução `CREATE NICKNAME` para especificar dados fixos a partir de um arquivo. Utilize a opção `DIRECTORY_PATH` para especificar dados fixos a partir de um diretório.

Quando os dados de origem são especificados enquanto a consulta está em execução, você pode utilizar o apelido para representar dados a partir de qualquer origem de documentos XML cujo esquema corresponde à definição do apelido. Estes documentos XML incluem:

Identificadores de Referência Uniforme

Um arquivo XML remoto ao qual uma URI faz referência fornece os dados do documento XML para o apelido. Especifique esta origem de documentos utilizando a opção de coluna de apelidos `DOCUMENT 'URI'`.

Colunas Relacionais

As colunas de uma tabela relacional, visualização ou apelido são utilizadas como entrada para seu documento XML. Especifique esta origem de documentos utilizando a opção de coluna de apelidos `DOCUMENT 'COLUMN'`.

Arquivo

Um único arquivo que contém dados XML é fornecido como entrada enquanto a consulta é executada. Especifique esta origem de documentos utilizando a opção de coluna de apelidos `DOCUMENT 'FILE'`.

Diretório

Vários arquivos XML em um caminho de diretório especificado fornecem os dados enquanto a consulta é executada. Especifique esta origem de documentos utilizando a opção de coluna de apelidos DOCUMENT 'DIRECTORY'.

Especifique a opção de coluna DOCUMENT para indicar que os dados da origem são fornecidos no tempo de consulta. Especifique URI, COLUMN, FILE ou DIRECTORY com a coluna DOCUMENT para indicar o tipo de origem de documentos XML.

Você não pode especificar uma opção FILE_PATH ou uma opção DIRECTORY_PATH com uma opção de coluna DOCUMENT.

Utilize a opção STREAMING para separar os dados do documento XML em fragmentos. Você pode utilizar a opção STREAMING com dados que estão em um formato fixo ou dados que são de nomes de origens que você especifica ao executar uma consulta. O wrapper XML processa o fluxo resultante de dados XML e extrai as informações que são solicitadas por um fragmento de consulta. O wrapper XML analisa um fragmento por vez. Como os fragmentos são analisados um por vez, o total da utilização de memória é reduzido. O tempo de processamento necessário para executar toda a consulta aumenta dependendo da capacidade de memória do seu servidor. Portanto, utilize a opção STREAMING apenas para analisar grandes documentos XML com 50 megabytes ou mais.

Você também pode escolher valores de opções de apelidos que ajudam a otimizar consultas que recuperam grandes quantidades de dados XML ou dados que contêm vários elementos aninhados. Estas opções são:

- INSTANCE_PARSE_TIME
- XPATH_EVAL_TIME
- NEXT_TIME

Você pode configurar valores para estas opções para testar e otimizar a consulta XML. Estes valores de opções controlam o tempo de processamento necessário para localizar elementos e analisar os dados nas linhas do documento XML.

O recurso de modelo de custo para o wrapper XML:

O wrapper XML fornece um recurso de modelo de custo para otimizar consultas em apelidos que correspondem aos seus documentos de origem XML.

Ao criar um apelido utilizando a instrução CREATE NICKNAME, você pode especificar as seguintes opções de apelidos para suportar o recurso de modelo de custo:

- INSTANCE_PARSE_TIME
- XPATH_EVAL_TIME

Você pode utilizar os valores padrão para estas opções de apelidos. Ou você pode configurar os valores para estas opções de apelidos para otimizar consultas nos apelidos raiz e não-raiz que você criar.

A opção de apelido INSTANCE_PARSE_TIME é a quantidade de tempo, em milissegundos, necessária para ler e analisar um elemento-raiz da linha de produção do apelido raiz. O tempo de análise inclui todos os elementos não-raiz contidos na linha de produção. Por exemplo, se o apelido raiz for customers, os

elementos não-raiz serão todos os elementos que correspondem aos orders, payments e items de cada cliente. O wrapper XML constrói uma estrutura na memória para representar estes elementos-raiz e não-raiz da linha de produção.

A opção de apelido XPATH_EVAL_TIME é a quantidade de tempo, em milissegundos, necessária para avaliar as expressões XPath que localizam os dados correspondentes a uma linha do apelido. As expressões XPath que são avaliadas incluem as expressões XPath que localizam as linhas atuais e as expressões XPath que localizam valores de colunas dentro destas linhas.

Espaços de Nomes para Origens de Dados XML:

Utilize a opção de coluna de apelidos NAMESPACES para identificar os elementos ou atributos que são parte de um espaço de nomes.

Você pode especificar a opção de apelido NAMESPACES ao registrar apelidos. O valor da opção de colunas de apelidos NAMESPACES é uma lista separada por vírgulas dos pares de nome e valor. O wrapper XML utiliza os pares de nome e valor para resolver os prefixos de espaços de nomes que estão nas expressões XPath de apelidos. Os prefixos que são utilizados nas expressões XPath são processados pelo processador XPath.

No exemplo a seguir, o documento XML inclui informações de nome, código e descrição de três produtos. O documento XML declara dois espaços de nomes, `http://www.one.com` e `http://www.two.com`, e possui um espaço de nomes padrão `http://www.default.com`. O elemento `product` está associado ao espaço de nomes `ns1`. O elemento `product` contém os atributos `name` e `code` e o elemento `desc`. O atributo `name` não está associado a um espaço de nomes. O atributo `code` está associado ao espaço de nomes `ns2`. O elemento `desc` está associado ao espaço de nomes `default`.

```
<?xml version="1.0" encoding="UTF-8"?>
<doc xmlns:ns1="http://www.one.com" xmlns:ns2="http://www.two.com"
  xmlns="http://www.default.com">
<ns1:product name="Computer" ns2:code="ABC123"
  <desc>"The Computer product description"</desc>
<ns1:product name="Keyboard" ns2:code="EFG456"
  <desc>"The Keyboard product description"</desc>
<ns1:product name="Mouse" ns2:code="HIJ789"
  <desc>"The Mouse product description"</desc>
</ns1:product>
</doc>
```

A tabela a seguir mostra o espaço de nomes que está associado a cada elemento e atributo no documento XML.

Tabela 43. Os elementos, atributos e espaços de nomes no documento XML

Elemento ou atributo	Espaço de nomes no documento XML	Notas
product: Um elemento no documento XML.	ns1="http://www.one.com"	Nenhum
name: Um atributo do elemento product no documento XML.	Nenhum	O atributo name não está associado a um espaço de nomes.
code: Um atributo do elemento product no documento XML.	ns2="http://www.two.com"	Nenhum

Tabela 43. Os elementos, atributos e espaços de nomes no documento XML (continuação)

Elemento ou atributo	Espaço de nomes no documento XML	Notas
desc: Um elemento dentro do elemento product no documento XML.	"http://www.default.com"	O elemento desc utiliza o espaço de nomes default, o qual não contém um prefixo.

Ao registrar o apelido para o documento XML, você define três colunas para corresponder aos elementos e atributos no documento XML. Você especifica as informações de espaço de nomes na opção de apelido NAMESPACES. Por Exemplo:

```
CREATE NICKNAME products
(name VARCHAR(16) OPTIONS (XPATH '@name'),
code VARCHAR(16) OPTIONS (XPATH '@pre2:code')
description VARCHAR (256) OPTIONS (XPATH './default:desc'))
FOR SERVER xml_server
OPTIONS (FILE_PATH '/home/mbreining/sql/xml/namespaces.xml',
XPATH '//pre1:name',
NAMESPACES 'pre1="http://www.one.com", pre2="http://www.two.com",
default="http://www.default.com");
```

A tabela a seguir mostra como os espaços de nomes do documento XML correspondem aos valores que você especifica na instrução CREATE NICKNAME.

Tabela 44. Espaços de nomes do documento XML e valores correspondentes na instrução CREATE NICKNAME.

Espaço de nomes no documento XML	Nome da coluna na instrução CREATE NICKNAME	Valor na opção de coluna XPATH	Valor na opção de apelido NAMESPACES
O espaço de nomes é ns1="http://www.one.com".	Nenhum	Nenhum	O valor é pre1="http://www.one.com". O valor é um prefixo que você define (pre1) e o identificador exclusivo para o espaço de nomes ("http://www.one.com").
Nenhum(a). O atributo não está associado a um espaço de nomes.	Nome	O valor é @name. O valor é um atributo que está no documento XML (name).	Nenhum
O espaço de nomes é ns2="http://www.two.com"	Código	O valor é @pre2:code. O valor é um prefixo que você define (pre2) e um atributo que está no documento XML (code).	O valor é pre2="http://www.two.com". O valor é um prefixo que você define (pre2) e o identificador exclusivo para o espaço de nomes ("http://www.two.com").
O espaço de nomes é "http://www.default.com" O espaço de nomes padrão não contém um prefixo.	Descrição	O valor é ./default:desc. O valor é um prefixo que você define (default) e um elemento que está no documento XML (desc).	O valor é default="http://www.default.com". O valor é um prefixo que você define (default) e o identificador exclusivo para o espaço de nomes ("http://www.default.com").

A opção de apelido NAMESPACES utiliza descritores compactados para suportar cadeias que são maiores do que 256 caracteres.

Para obter informações adicionais sobre espaços de nomes XML, consulte a explicação de espaços de nomes no Web site W3C.

Registrando Apelidos para Origens de Dados XML:

Para cada definição de servidor XML registrada, você deve registrar um apelido para cada documento XML que deseja acessar. Utilize estes apelidos em vez dos nomes de documentos XML quando consultar as origens de dados XML.

Sobre Esta Tarefa

Restrições

Se você tentar acessar as origens de dados XML que estão em uma unidade compartilhada a partir de um servidor federado que executa o Windows 2003, suas consultas podem falhar. Esta é uma limitação do Windows 2003. Você pode evitar este problema especificando o caminho absoluto na opção FILE_PATH na instrução CREATE NICKNAME.

Você deve criar apelidos que correspondem à estrutura em árvore de sua origem de dados XML. Os apelidos-pai correspondem a estrutura da raiz da árvore. Os apelidos-filho correspondem aos elementos aninhados dentro do elemento para o apelido-pai.

Procedimento

Emita a instrução CRIAR APELIDO a partir da linha de comandos. Os apelidos podem ter até 128 caracteres de comprimento.

Instrução CREATE NICKNAME - Exemplo para Wrapper XML:

Utilize a instrução CREATE NICKNAME para registrar apelidos para documentos XML. Existe um exemplo completo, o qual mostra como criar apelidos pai e filho, e exemplo para opções de colunas específicas.

Recomendação: Não utilize o operador descendant-or-self // ao especificar opções de colunas e apelidos XPath em suas consultas. O operador descendant-or-self é um operador XPath. A utilização do operador descendant-or-self pode reduzir o desempenho do servidor federado.

Exemplo Completo

O exemplo a seguir mostra como criar apelidos para origens de dados XML utilizando o arquivo XML de amostra mostrado no arquivo XML de amostra a seguir.

```

<?xml version="1.0" encoding="UTF-8"?>
<doc>
  <customer id='123'>
    <name>...</name>
    <address>...</address>
    ...
    <order>
      <amount>...</amount>
      <date>...</date>
      <item quant='12'>
        <name>...</name>
      </item>
      <item quant='4'>...</item>
      ...
    </order>
    <order>...</order>
    ...
    <payment>
      <number>...</number>
      <date>...</date>
    </payment>
    <payment>...</payment>
    ...
  </customer>
  <customer id='124'>...</customer>
</doc>

```

Figura 27. Arquivo XML de Amostra

O Apelido-pai

A primeira etapa é criar o apelido-pai, `customers`. Para criar o apelido, emita a seguinte instrução:

```

CREATE NICKNAME customers (
  id      VARCHAR(5)    OPTIONS(XPATH './@id')
  name    VARCHAR(16)   OPTIONS(XPATH './name'),
  address VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid     VARCHAR(16)   FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(DIRECTORY_PATH '/home/db2user',
        XPATH '/doc/customer', STREAMING 'YES');

```

Esta instrução cria o apelido `customers` em vários arquivos XML no caminho de diretório especificado, `/home/db2user`.

A opção de apelido `STREAMING` indica que os dados de origem XML são separados e processados pelo nó, neste exemplo, pelo registro do cliente. Quando a opção de apelido `STREAMING` é utilizada, o wrapper não armazena todo o documento XML na memória. Em vez disso, o wrapper XML divide o documento em várias seções que são analisadas individual e seqüencialmente. A opção de apelido `STREAMING` deve ser utilizada apenas com documentos XML grandes. O desempenho de suas consultas é impactado quando você utiliza esta opção.

Os Apelidos-filho

A próxima etapa é criar os apelidos-filho para os elementos `orders`, `payments` e `items`.

Emita a seguinte instrução para criar o apelido-filho `orders`:

```
CREATE NICKNAME orders(
  amount INTEGER      OPTIONS(XPATH './amount'),
  date   VARCHAR(10)  OPTIONS(XPATH './date'),
  oid    VARCHAR(16)  OPTIONS(PRIMARY_KEY 'YES'),
  cid    VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
FOR SERVER xml_server
OPTIONS( XPATH './order');
```

Emita a seguinte instrução para criar o apelido-filho payments:

```
CREATE NICKNAME payments(
  number INTEGER      OPTIONS(XPATH './number'),
  date   VARCHAR(10)  OPTIONS(XPATH './date'),
  cid    VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
FOR SERVER xml_server
OPTIONS( XPATH './payment');
```

Emita a seguinte instrução para criar o apelido-filho items:

```
CREATE NICKNAME items(
  name      VARCHAR(20)  OPTIONS(XPATH './name'),
  quantity  INTEGER      OPTIONS(XPATH './@quant'),
  oid       VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'ORDERS'))
FOR SERVER xml_server
OPTIONS( XPATH './item');
```

Exemplos de Opções de Colunas de Apelidos

Os exemplos a seguir mostram como incluir as opções de colunas de apelidos DOCUMENT ao criar apelidos. Os exemplos também mostram como essas opções são utilizadas em consultas.

Exemplo DOCUMENT 'FILE'

Os exemplos a seguir mostram como incluir as opções de colunas de apelidos DOCUMENT ao criar apelidos. Os exemplos também mostram como utilizar essas opções em consultas.

O exemplo CREATE NICKNAME a seguir mostra a utilização da opção de coluna de apelidos DOCUMENT 'FILE':

```
CREATE NICKNAME customers (
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid      VARCHAR(16)   FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');
```

Você pode, então, executar a seguinte consulta no apelido customers, especificando o local do documento XML na cláusula WHERE:

```
SELECT * FROM customers WHERE doc = '/home/db2user/Customers.xml';
```

Exemplo DOCUMENT 'DIRECTORY'

O exemplo CREATE NICKNAME a seguir mostra a utilização da opção de coluna de apelidos DOCUMENT 'DIRECTORY':

```
CREATE NICKNAME customers (
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'DIRECTORY'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
```



```

address VARCHAR(30)  OPTIONS(XPATH './address/@street'),
cid      VARCHAR(16) FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');

```

Você pode, então, executar a seguinte consulta no apelido customers:

```
SELECT name FROM customers WHERE doc = '/home/data/xml';
```

Esta consulta recupera os documentos XML localizados no caminho de diretório /home/data/xml, o qual é especificado na cláusula WHERE.

Exemplo DOCUMENT 'URI'

O exemplo CREATE NICKNAME a seguir mostra a utilização da opção de coluna de apelidos DOCUMENT 'URI':

```

CREATE NICKNAME customers (
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'URI'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid      VARCHAR(16)   FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');

```

Você pode, então, executar a seguinte consulta no apelido customers para recuperar os dados XML do local remoto:

```
SELECT * FROM customers WHERE doc = 'http://www.lg-mv.org/foo.xml';
```

Exemplo DOCUMENT 'COLUMN'

O exemplo CREATE NICKNAME a seguir mostra a utilização da opção de coluna de apelidos DOCUMENT 'COLUMN':

```

CREATE NICKNAME emp(
  doc      VARCHAR(500)  OPTIONS(DOCUMENT 'COLUMN')
  fname    VARCHAR(16)   OPTIONS(XPATH '@first'),
  lname    VARCHAR(16)   OPTIONS(XPATH '@last'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/name');

```

Você pode, então, executar uma das seguintes consultas no apelido emp para recuperar os dados XML:

```

SELECT * FROM emp WHERE doc = '<?xml version="1.0" encoding="UTF-8"?>
<doc>
  <title> employees </title>
  <name first="David" last="Marston"/>
  <name first="Donald" last="Leslie"/>
  <name first="Emily" last="Farmer"/>
  <name first="Myriam" last="Midy"/>
  <name first="Lee" last="Tran"/>
  <name first="Lili" last="Farmer"/>
  <name first="Sanjay" last="Kumar"/>
</doc>';

```

ou

```
SELECT * FROM emp WHERE doc = (SELECT * FROM xml_tab);
```

A tabela xml_tab contém uma coluna que é preenchida com os dados XML.

Consultas para Origens de Dados XML

Antes de criar consultas para acessar origens de dados XML, há ações que você deve tomar para otimizar o desempenho da consulta.

Visualizações Federadas

Você pode utilizar visualizações federadas para garantir que as consultas que juntam as partes de uma hierarquia de apelidos XML sejam executadas corretamente.

Evitar Operador descendant-or-self

Não utilize o operador descendant-or-self // ao especificar colunas XPATH e opções de apelidos ao criar os apelidos XML. O operador descendant-or-self é um operador XPath, e a utilização do operador descendant-or-self pode reduzir o desempenho do servidor federado.

Servidores Federados Windows 2003

Se você tentar acessar as origens de dados XML que estão em uma unidade compartilhada em um servidor federado que executa o Windows 2003, sua consulta pode falhar com a seguinte mensagem de erro:

```
SQL1822N Código de erro inesperado "ERRNO = 2" recebido da origem de dados
"XML_SERVER". O texto e tokens associados são "Não foi possível ler o arquivo".
SQLSTATE=560BD
```

Esta é uma limitação do Windows 2003. Você pode evitar este problema especificando o caminho absoluto na opção FILE_PATH na instrução CREATE NICKNAME.

O exemplo a seguir mostra uma instrução CREATE NICKNAME com um caminho abreviado especificado na opção FILE_PATH:

```
CREATE NICKNAME customers
(
  id      VARCHAR(5)  OPTIONS(XPATH './@id'),
  name    VARCHAR(16) OPTIONS(XPATH './name'),
  address VARCHAR(30) OPTIONS(XPATH './address/@street'),
  cid     VARCHAR(16) FOR BIT DATA NOT NULL
          OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
  OPTIONS(DIRECTORY_PATH '\home\db2user',
          XPATH '/doc/customer', STREAMING 'YES');
```

As consultas que utilizam este apelido podem falhar pois você especificou o caminho abreviado.

Para servidores federados que executando o Windows 2003, especifique o caminho absoluto na opção FILE_PATH na instrução CREATE NICKNAME.

Por exemplo:

```
CREATE NICKNAME customers
(
  id      VARCHAR(5)  OPTIONS(XPATH './@id'),
  name    VARCHAR(16) OPTIONS(XPATH './name'),
  address VARCHAR(30) OPTIONS(XPATH './address/@street'),
  cid     VARCHAR(16) FOR BIT DATA NOT NULL
          OPTIONS(PRIMARY_KEY 'YES'))
```

```
FOR SERVER xml_server
OPTIONS(DIRECTORY_PATH '\\host.svl.ibm.com\D$\home\db2user',
XPATH '/doc/customer', STREAMING 'YES');
```

Criando Visualizações Federadas para os Apelidos do Wrapper XML

Você pode criar visualizações federadas na hierarquia de apelidos que descreve um documento XML. A definição de visualizações federadas garante que as consultas que juntam as partes de uma hierarquia de apelidos XML sejam executadas corretamente.

Sobre Esta Tarefa

Sobre esta tarefa

Uma *visualização federada* é uma visualização no banco de dados federado que faz referência a um apelido, em vez de uma tabela de origens de dados.

Na hierarquia de apelidos XML, o apelido raiz e as consultas que juntam colunas, diferente das colunas especiais PRIMARY_KEY e FOREIGN_KEY, não são impactados pela utilização de visualizações federadas.

Quando você cria visualizações federadas para apelidos XML, você deve incluir todos os predicados necessários e um caminho completo para o diretório raiz.

Procedimento

Para criar visualizações federadas para apelidos XML:

Procedimento

Utilize a instrução CREATE VIEW para definir uma visualização para cada apelido não-raiz. A visualização deve ser uma junção de todos os apelidos no caminho para o apelido raiz.

1. Na cláusula WHERE da visualização, defina as colunas PRIMARY_KEY e FOREIGN_KEY como os predicados de junção.
2. Na lista SELECT, inclua todas as colunas do apelido não-raiz, exceto a coluna que é designada com a opção de coluna de apelido FOREIGN_KEY. Na lista SELECT, inclua a coluna do apelido-pai designado com a opção PRIMARY_KEY.

Instrução CREATE VIEW - Exemplos para o Wrapper XML

Utilize a instrução CREATE VIEW para criar visualizações federadas para apelidos não-root. Este exemplo inclui um arquivo XML de amostra, as instruções que você utiliza para criar as visualizações, e mostra como você pode utilizar as visualizações em uma consulta.

Você pode criar visualizações federadas na hierarquia de apelidos que descrevem um documento XML para garantir que as consultas que juntam partes de uma hierarquia de apelidos XML sejam executadas corretamente. Ao especificar uma visualização federada em uma consulta, os dados são recuperados da origem de dados remota.

Os exemplos a seguir mostram como criar visualizações federadas para apelidos não-root para descrever documentos de origem XML.

Arquivo XML de Amostra

Os exemplos a seguir têm como base este arquivo XML de amostra.

```
<?xml version="1.0" encoding="UTF-8"?>
<doc>
  <customer id='123'>
    <name>...</name>
    <address>...</address>
    ...
    <order>
      <amount>...</amount>
      <date>...</date>
      <item quant='12'>
        <name>...</name>
      </item>
      <item quant='4'>...</item>
      ...
    </order>
    <order>...</order>
    ...
    <payment>
      <number>...</number>
      <date>...</date>
    </payment>
    <payment>...</payment>
    ...
  </customer>
  <customer id='124'>...</customer>
</doc>
```

Figura 28. Arquivo XML de Amostra

Instruções CREATE VIEW para os Apelidos Não-root

O exemplo a seguir mostra como criar uma visualização federada para o apelido não-raiz order:

```
CREATE VIEW order_view AS
  SELECT o.amount, o.date, o.oid, c.cid
  FROM customers c, orders o
  WHERE c.cid = o.cid;
```

O exemplo a seguir mostra como criar uma visualização federada para o apelido não-raiz payment:

```
CREATE VIEW payment_view AS
  SELECT p.number, p.date, c.cid
  FROM customers c, payments p
  WHERE c.cid = p.cid;
```

O exemplo a seguir mostra como criar uma visualização federada para o apelido não-raiz item:

```
CREATE VIEW item_view AS
  SELECT i.quantity, i.name, o.oid
  FROM customers c, orders o, items i
  WHERE c.cid = o.cid AND o.oid = i.oid;
```

Uma consulta que utiliza as visualizações federadas

As consultas que são submetidas para as visualizações federadas são processadas corretamente porque o caminho de junção para o diretório root é especificado na cláusula WHERE.

Por exemplo, a consulta a seguir utiliza o número de identificação do cliente e a data que um pedido foi realizado para retornar a quantidade que foi pedida e o valor do pagamento devido. Em vez de utilizar os apelidos na consulta, as visualizações são especificadas na cláusula FROM.

```
SELECT o.amount, p.amount
FROM order_view o, payment_view p
WHERE p.date = o.date AND
      p.cid = o.cid;
```

Dicas de Otimização de Consultas para o Recurso de Modelo de Custo XML

O recurso de modelo de custo para o wrapper XML ajuda a otimizar consultas nos apelidos que você criar.

O recurso de modelo de custo utiliza as seguintes opções de apelidos da instrução CREATE NICKNAME:

- INSTANCE_PARSE_TIME
- XPATH_EVAL_TIME

Você pode especificar valores para estas opções de apelidos ao emitir uma instrução CREATE NICKNAME para registrar um apelido para uma origem de dados XML.

O recurso de modelo de custo utiliza estes valores de parâmetros para determinar a quantidade de tempo necessária para analisar os dados em cada linha de um documento de origem XML. Os valores de parâmetros também são utilizados para avaliar a expressão XPath para o apelido.

Você pode utilizar os valores padrão para estas opções de apelidos. Entretanto, se desejar otimizar consultas em estruturas grandes e complexas de origem XML para os apelidos que criar, utilize o exemplo a seguir como um guia.

Um exemplo de otimização de uma consulta grande

Seu documento XML possui um esquema relacional com quatro apelidos:

- customers
- orders
- payments
- items

O apelido raiz é customers.

Execute consultas em cada apelido. Execute cada consulta em uma amostra dos dados XML que são típicos em seu ambiente.

Por exemplo:

```
SELECT * from customers;
SELECT * from orders;
SELECT * from payments;
SELECT * from items;
```

Tome nota da quantidade de tempo (em milissegundos) que é necessária para executar cada consulta utilizando o comando **db2batch**, ou um comando ou utilitário equivalente. Você pode utilizar o comando **db2batch** para obter um arquivo de saída que contém o tempo necessário para executar as consultas. Tome nota do número de tuplas que são retornadas.

Par cada apelido, utilize as seguintes fórmulas para determinar os valores ideais para as opções de apelidos `INSTANCE_PARSE_TIME` e `XPATH_EVAL_TIME`:

$$\text{INSTANCE_PARSE_TIME} = (75\% \times \text{run time of SELECT * query}) \div \text{number of tuples returned}$$

$$\text{XPATH_EVAL_TIME} = (25\% \times \text{run time of SELECT * query}) \div \text{number of tuples returned}$$

Para o apelido raiz (neste exemplo, `customers`), utilize os valores calculados para as opções de apelidos `INSTANCE_PARSE_TIME` e `XPATH_EVAL_TIME`.

Para apelidos não-raiz, (neste exemplo, `orders`, `payments` e `items`), utilize apenas o valor calculado para o parâmetro `XPATH_EVAL_TIME`. O valor do parâmetro `INSTANCE_PARSE_TIME` não é aplicável para apelidos não-raiz.

Você pode utilizar estas fórmulas como um guia para ajuste de suas consultas. Os valores ideais para estas opções de apelidos também dependem da complexidade de seus documentos de origem XML e da velocidade do processador que está utilizando.

Origem de Dados XML - Consultas de Exemplo

Exemplos de consultas utilizando apelidos XML.

Estes exemplos utilizam os apelidos de clientes, pedidos e itens.

Uma Consulta que Retorna um Valor Específico dos Documentos XML

Quando a seguinte instrução `SELECT` é executada, o wrapper retorna os nomes de todos os clientes:

```
SELECT name FROM customers;
```

Uma Consulta que Retorna Todos os Registros de um Determinado Cliente

Quando a seguinte instrução `SELECT` é executada, o wrapper retorna todos os registros nos quais o nome do cliente é `Chang`:

```
SELECT * FROM customers
WHERE name='Chang';
```

Uma consulta que retorna valores específicos com base em uma condição de junção entre um apelido pai e filho

Quando a seguinte instrução `SELECT` é executada, o wrapper retorna os nomes e quantidades de clientes para cada pedido realizado por cada cliente. Você deve especificar a junção, `c.cid=o.cid`, para indicar o relacionamento pai-filho entre o apelido `customers` e o apelido `orders`.

```
SELECT c.name, o.amount FROM customers c, orders o
WHERE c.cid=o.cid;
```

Uma consulta que mostra como especificar condições de junção entre um apelido-pai e diversos apelidos-filho

Quando a seguinte instrução SELECT é executada, o wrapper retorna os endereços de clientes, quantidades de pedidos e nomes de itens para cada pedido e item de cada cliente. Você deve especificar as duas junções para manter os relacionamentos pai-filho.

```
SELECT c.address, o.amount, i.name FROM customers c, orders o, items i
      WHERE c.cid=o.cid AND o.oid=i.oid;
```

Consultas que mostram como especificar um documento XML em uma consulta

Os exemplos a seguir mostram como escrever consultas utilizando um apelido que especifica uma opção de coluna de apelido DOCUMENT em vez de uma opção de apelido FILE_PATH.

A instrução CREATE NICKNAME que é utilizada para criar o apelido customers é a seguinte:

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid      VARCHAR(16)   OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');
```

A consulta a seguir seleciona todos os dados do arquivo XML Customers.xml com um caminho de /home/db2user/Customers.xml:

```
SELECT * FROM customers
      WHERE doc='/home/db2user/Customers.xml';
```

A consulta a seguir seleciona os nomes de clientes e as datas de seus pedidos a partir do arquivo Customers.xml para cada pedido com uma quantidade acima de 1000. O caminho /home/db2user/Customers.xml especifica o local do arquivo Customers.xml.

```
SELECT c.name, o.date FROM customers c, orders o
      WHERE c.doc='/home/db2user/Customers.xml' AND o.amount > 1000;
```

Capítulo 3. Suporte de origem de dados para recursos federados

Consulte esta tabela quando desejar saber se uma origem de dados suporta ou não um recurso federado específico.

Antes de ser possível utilizar alguns desses recursos, é possível que seja necessário configurar as opções de wrapper ou servidor específicos ou executar outras tarefas para ativar a funcionalidade. Para obter mais informações, consulte os tópicos específicos em cada recurso.

Tabela 45. Recursos e Origens de Dados Suportadas

Recurso	Origens de Dados
Pontos de salvamento do aplicativo com operações WRITE em relação aos apelidos	DB2 for Linux, UNIX e Windows
Otimização de assincronia	Todas as origens de dados
Tabelas de Caches	Família DB2 Informix Microsoft SQL Server Oracle Sybase
Importação de dados nos apelidos	Família DB2 Informix Microsoft SQL Server Oracle Sybase Teradata
Tolerância de erros em expressões de tabelas aninhadas	Família DB2 Informix JDBC Microsoft SQL Server ODBC Oracle Sybase Teradata
Repositório de mapeamento de usuário externo	Todas as origens de dados
Indicadores de funcionamento federados	Família DB2 Excel Informix JDBC Microsoft SQL Server ODBC Oracle Sybase Arquivos estruturados em tabelas Teradata XML (apenas os apelidos raiz)

Tabela 45. Recursos e Origens de Dados Suportadas (continuação)

Recurso	Origens de Dados
Procedimentos federados	Família DB2, no modo confiável Oracle, no modo confiável Microsoft SQL Server, no modo confiável Sybase, no modo em fence, com o servidor federado instalado no UNIX Sybase, no modo em fence, com o servidor federado instalado no Linux ou Microsoft Windows
Contextos confiáveis federados	DB2 para Linux, UNIX e Windows Versão 9.5 DB2 para z/OS Versão 9 Oracle
proxy HTTP	serviços da Web XML
Isolamento de nível de conexão	Família DB2 Informix JDBC Microsoft SQL Server ODBC Oracle Sybase
Controle de acesso baseado em etiqueta	DB2 forLinux, UNIX e Windows Versão 9.1 e 9.5 Oracle
Operações de leitura e gravação de LOB	DB2 para z/OS DB2 para Linux, UNIX e Windows DB2 para System i Oracle Teradata
Operações somente leitura de LOB	BioRS Informix JDBC Microsoft SQL Server ODBC Script Sybase serviços da Web XML
Tabelas de consultas materializadas	Todas as origens de dados, com restrições específicas
Recurso de Atualização de Estatísticas de Pseudônimo	BioRS Família DB2 Excel Informix JDBC Microsoft SQL Server ODBC Oracle Sybase Arquivos estruturados em tabelas Teradata XML (apenas os apelidos raiz)

Tabela 45. Recursos e Origens de Dados Suportadas (continuação)

Recurso	Origens de Dados
Sessões de passagem	DRDA Informix Oracle Microsoft SQL Server Sybase Teradata
Tipo de dados remotos do XML	DB2 para Linux, UNIX e Windows wrapper do XML
proxy SOCKS	BioRS Script serviços da Web XML All relational data sources Nota: To connect to relational data sources, you must set DB2ENVLIST=SOCKS5C_CONFIG.
Secure Socket Layer (SSL)	serviços da Web XML
Isolamento de nível de instrução	Família DB2 Microsoft SQL Server
Transações two-phase commit	DB2 para Linux, UNIX e Windows, no modo confiável DB2 para System i, no modo confiável DB2 para z/OS, no modo confiável Informix, no modo confiável Microsoft SQL Server, no modo confiável, com o servidor federado instalado no Microsoft Windows Oracle, no modo confiável Sybase, no modo confiável, com o servidor federado instalado no Microsoft Windows Sybase, no modo em fence, com o servidor federado instalado no UNIX
Suporte a Unicode	Todas as origens de dados

Capítulo 4. Referência de Opções de Origens de Dados

Cada origem de dados suporta opções específicas de wrappers, servidores, mapeamentos de usuários, apelidos e colunas.

Referência de Opções do BioRS

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, servidor, mapeamento de usuário, apelido e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas nas instruções CREATE WRAPPER e CREATE SERVER.

Tabela 46. Opções de Wrapper para BioRS

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.
PROXY_SERVER_NAME	Especifica o nome ou endereço IP do servidor proxy. Esta opção é necessária se o valor de PROXY_TYPE for HTTP ou SOCKS. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.
PROXY_SERVER_PORT	Especifica a porta ou o nome do serviço para o serviço de proxy no servidor proxy. Esta opção é necessária se o valor de PROXY_TYPE for HTTP ou SOCKS. Os valores válidos são um número da porta decimal de 1 a 32760 ou um nome do serviço.

Tabela 46. Opções de Wrapper para BioRS (continuação)

Nome	Descrição
PROXY_TYPE	Especifica o tipo de proxy a ser utilizado para acessar a Internet, quando o servidor federado está protegido por firewall. Os valores válidos são NONE, HTTP e SOCKS. O valor-padrão é NONE. Se você configurar esta opção como HTTP ou SOCKS, é necessário também especificar o PROXY_SERVER_NAME e o PROXY_SERVER_PORT.

Opções do Servidor

Tabela 47. Opções de Servidor para BioRS

Nome	Descrição
CASE_SENSITIVE	Especifica se o servidor BioRS trata os nomes de forma a fazer distinção entre maiúsculas e minúsculas. Os valores válidos são Y e N. O padrão é Y; os nomes são tratados em uma forma a fazer distinção entre maiúsculas e minúsculas. No produto BioRS, um parâmetro de configuração controla a distinção entre maiúsculas e minúsculas dos dados armazenados no servidor BioRS. A opção CASE_SENSITIVE e o parâmetro de configuração devem especificar a mesma distinção entre maiúsculas e minúsculas. Se for necessário alterar o valor da opção CASE_SENSITIVE após criar a definição de servidor, é necessário descartar a definição de servidor e criar a definição e todos os apelidos novamente.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma para o plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Tabela 47. Opções de Servidor para BioRS (continuação)

Nome	Descrição
NODE	Necessário. Especifica o nome do host DNS ou endereço IP do sistema no qual a ferramenta de consulta BioRS está disponível. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado. O valor-padrão é localhost.
PORT	Especifica a porta para conexões ao servidor BioRS. Os valores válidos são uma porta numérica ou um nome do serviço TCP/IP. The default is 5014.
PROXY_AUTHID	Especifica o nome de usuário para a autenticação do servidor proxy.
PROXY_PASSWORD	Especifica a senha para a autenticação do servidor proxy.
PROXY_SERVER_NAME	Especifica o nome ou endereço IP do servidor proxy. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.
PROXY_SERVER_PORT	Especifica a porta ou o nome do serviço para o serviço de proxy no servidor proxy. Os valores válidos são um número da porta decimal de 1 a 32760 ou um nome do serviço.
PROXY_TYPE	Especifica o tipo de proxy a ser utilizado para acessar a Internet, quando o servidor federado está protegido por firewall. Os valores válidos são NONE, HTTP e SOCKS. O valor-padrão é NONE.
TIMEOUT	Especifica o tempo máximo, em minutos, que o servidor federado aguarda por uma resposta do servidor remoto. O padrão é 10.

Opções de Mapeamento de Usuários

Tabela 48. Opções de Mapeamento de Usuários para BioRS

Opção	Descrição
GUEST	Especifica se o ID de autenticação GUEST BioRS é utilizado para executar operações. Os valores válidos são Y e N. O padrão é N; o ID de GUEST BioRS não é utilizado. Esta opção não é válida, se você especificar as opções REMOTE_AUTHID e REMOTE_PASSWORD.
PROXY_AUTHID	Especifica o nome de usuário para a autenticação do servidor proxy.

Tabela 48. Opções de Mapeamento de Usuários para BioRS (continuação)

Opção	Descrição
PROXY_PASSWORD	Especifica a senha para a autenticação do servidor proxy. A senha é criptografada quando é armazenada no catálogo de banco de dados federado.
REMOTE_AUTHID	Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.

Opções de Apelido

Tabela 49. Opções de Apelidos para BioRS

Opção	Descrição
REMOTE_OBJECT	Especifica o nome do banco de dados BioRS associado ao apelido. Este nome determina o esquema e o banco de dados BioRS para o apelido. O nome também especifica o relacionamento do apelido com outros apelidos. A distinção entre maiúsculas e minúsculas desta opção depende da distinção entre maiúsculas e minúsculas do servidor BioRS e do valor da opção do servidor CASE_SENSITIVE. Não é possível utilizar a instrução ALTER NICKNAME para alterar ou excluir este nome. Se o nome do banco de dados BioRS é alterado, é necessário excluir o apelido e, em seguida, criá-lo novamente.
TIMEOUT	Especifica o tempo máximo, em minutos, a aguardar pela resposta do servidor de origem de dados. O padrão é 10.

Opções de Coluna

Tabela 50. Opções de Colunas para BioRS

Opção	Descrição
ELEMENT_NAME	Especifica o nome de elemento do BioRS. A distinção entre maiúsculas e minúsculas deste nome depende da distinção entre maiúsculas e minúsculas do servidor BioRS e do valor da opção CASE_SENSITIVE do servidor. É necessário especificar o nome de elemento do BioRS apenas se for diferente do nome da coluna.

Tabela 50. Opções de Colunas para BioRS (continuação)

Opção	Descrição
IS_INDEXED	Especifica se a coluna correspondente é indexada e, portanto, pode ser referenciada em um predicado. Os valores válidos são Y e N. O padrão é N; a coluna não é indexada.
REFERENCED_OBJECT	Especifica o nome do banco de dados BioRS referenciado pela coluna atual. A distinção entre maiúsculas e minúsculas deste nome depende da distinção entre maiúsculas e minúsculas do servidor BioRS e do valor da opção CASE_SENSITIVE do servidor. Esta opção é válida apenas para colunas que possuem o tipo de dado BioRS de referência.

Referência das Opções do Banco de Dados DB2

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, servidor, mapeamento de usuário, apelido e as opções de coluna.

Opções de Wrapper

The following tables list the options that apply to DB2 data sources and identify the required options that you must specify.

Tabela 51. Opções de Wrapper para as Origens de Dados do DB2

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Opções do Servidor

Tabela 52. Opções de Servidor para as Origens de Dados do DB2

Nome	Descrição
APP_ISOLATION_ENABLE	Specifies the isolation level for a client connection. Valid values are Y and N. The default is N; the federated server does not set the isolation level for a connection to the client. In this case, the client can use the isolation level set in configuration file, for example db2cli.ini. Y specifies that the federated server obtains the isolation level from the application and sets the application isolation as a connection attribute.
COLLATING_SEQUENCE	Especifica se a origem de dados especifica a mesma seqüência de intercalação padrão que o banco de dados federado. Valid values are Y, N, and I. I specifies it is not case-sensitive. O padrão é Y. A seqüência de intercalação especificada para o o servidor federado deve corresponder à seqüência de intercalação na origem de dados remota.
COMM_RATE	Especifica a taxa de comunicação, em megabytes por segundo, entre o servidor federado e o servidor de origem de dados. Os valores válidos são números inteiros maiores que 0 e menores que 2147483648. O padrão é 2.
CPU_RATIO	Especifica o quão mais rápida ou mais lenta a CPU de origem de dados é em comparação à CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de CPU; uma proporção 1:1. Uma configuração 0.5 indica que a CPU do servidor federado é 50% mais lenta que a CPU da origem de dados. Uma configuração 2 indica que a CPU federada é duas vezes mais rápida que a CPU da origem de dados.
DATE_COMPAT	Especifica se o parâmetro date_compat é aplicado ao banco de dados. Os valores válidos são Y e N. O padrão é N. Esta opção do servidor é válida apenas para DB2 Database para Linux, UNIX e Windows Versão 9.7 e posterior. In InfoSphere Federation Server Version 9.7 Fix Pack 2 and later, when you run the CREATE SERVER statement, this server option is automatically configured based on the configuration of your data source. If you attempt to manually configure this server option, you receive the SQL1841N message.

Tabela 52. Opções de Servidor para as Origens de Dados do DB2 (continuação)

Nome	Descrição
DBNAME	Necessário. Especifica o banco de dados específico para uso para a conexão com o banco de dados DB2 inicial e remota. Este banco de dados específico é o alias do banco de dados para o banco de dados remoto DB2, catalogado no servidor federado utilizando o comando CATALOG DATABASE ou o assistente de configuração do DB2.
DB2_MAXIMAL_PUSHDOWN	Especifica os critérios primários que o otimizador de consultas utiliza para escolher um plano de acesso. Os valores válidos são Y e N. O padrão é N; o otimizador de consultas escolhe o plano que possui o menor custo estimado. Y especifica que o otimizador de consultas escolhe o plano de acesso que direciona a maioria das operações de consulta à origem de dados.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
DB2_TWO_PHASE_COMMIT	Especifica se o servidor federado se conecta à origem de dados no protocolo two-phase commit ou no protocolo one-phase commit. Os valores válidos são Y e N. O padrão é N; o servidor federado utiliza o protocolo one-phase commit para se conectar. Y especifica que o servidor federado utiliza o protocolo two-phase commit para se conectar.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Tabela 52. Opções de Servidor para as Origens de Dados do DB2 (continuação)

Nome	Descrição
FED_PROXY_USER	<p>Especifica o ID de autorização utilizado para estabelecer todas as conexões confiáveis de saída, quando a conexão de entrada não é confiável. The user whose ID is specified in this option must have a user mapping that specifies both the REMOTE_AUTHID and REMOTE_PASSWORD options.</p> <p>Restrição: This server option is only valid for DB2 Database para Linux, UNIX e Windows Version 9.5 and later and DB2 for z/OS Version 9 and later.</p>
FOLD_ID	<p>Especifica o caso para o ID do usuário que é enviado à origem de dados. There is no default value. The federated server sends the user ID in uppercase; then if the uppercase user ID fails, the server sends the user ID in lowercase. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.</p>
FOLD_PW	<p>Especifica o caso para a senha que é enviada à origem de dados. Não há valor-padrão; o servidor federado envia a senha em maiúsculas; em seguida, se a senha em maiúsculas falhar, o servidor envia a senha em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.</p>
IO_RATIO	<p>Especifica o quão mais rápido ou mais lento o sistema I/O de origem de dados é executado em comparação ao sistema I/O do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23}. O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de I/O; uma proporção 1:1. Uma configuração 0.5 indica que a velocidade do servidor federado é 50% mais lenta que a velocidade da origem de dados. Uma configuração 2 indica que a velocidade federada é duas vezes mais rápida que a velocidade da origem de dados.</p>

Tabela 52. Opções de Servidor para as Origens de Dados do DB2 (continuação)

Nome	Descrição
NO_EMPTY_STRING	<p>Especifica se o servidor de origem de dados remoto pode conter cadeias vazias. Os valores válidos são Y e N. O valor-padrão varia dependendo da origem de dados remota. Para as origens de dados remotas do Oracle, o padrão é Y; todos os valores de cadeias vazias são convertidos em valores NULL. Para todas as outras origens de dados remotas, o padrão é N; a origem de dados pode conter cadeias vazias.</p> <p>É possível melhorar o desempenho dos sistemas configurando esta opção como Y nas configurações do sistema, onde o servidor federado está no modo compatível VARCHAR2, mas a origem de dados remota não é compatível com VARCHAR2.</p>
NUMBER_COMPAT	<p>Especifica se o servidor de origem de dados suporta o tipo de dado NUMBER. Os valores válidos são Y e N. O padrão é N; o servidor de origem de dados não suporta o tipo de dado NUMBER. Nos sistemas onde o servidor federado não suporta o tipo de dado NUMBER, mas o servidor de origem de dados sim, é necessário configurar a opção NUMBER_COMPAT como Y, pois o servidor de origem de dados pode retornar resultados DECFLOAT, que são externos ao intervalo do tipo de dado DECIMAL e causar o erro SQLSTATE 560BD.</p> <p>Restrição: Esta opção do servidor é válida apenas para DB2 Database para Linux, UNIX e Windows Versão 9.5 e posterior e DB2 para z/OS Versão 9 e posterior.</p> <p>In InfoSphere Federation Server Version 9.7 Fix Pack 2 and later, when you run the CREATE SERVER statement, this server option is automatically configured based on the configuration of your data source. If you attempt to manually configure this server option, you receive the SQL1841N message.</p>
OLD_NAME_GEN	<p>Especifica como converter os nomes das colunas e os nomes de índices que estão na origem de dados nos nomes das colunas de apelidos e nomes de índices locais para o servidor federado. Os valores válidos são Y e N. O padrão é N; os nomes gerados correspondem fortemente aos nomes na origem de dados. Y especifica que os nomes gerados são os mesmos nomes criados no IBM WebSphere Federation Server Versão 9 e anterior. Assim, os nomes podem não corresponder tão fortemente aos nomes da origem de dados.</p>

Tabela 52. Opções de Servidor para as Origens de Dados do DB2 (continuação)

Nome	Descrição
PUSHDOWN	<p>Especifica se o servidor federado permite que a origem de dados avalie as operações. Os valores válidos são Y e N. O padrão é Y; a origem de dados avalia as operações. N especifica que o servidor federado envia instruções SQL que incluem apenas SELECT com os nomes das colunas. Predicados, como a coluna WHERE=;, as funções escalares, como MAX e MIN;,, as classificações, como ORDER BY OR GROUP BY;,, e as junções não são incluídas no SQL que o servidor federado envia à origem de dados.</p>
SAME_DECFLT_ROUNDING	<p>Especifica se o modo circular do servidor federado e do servidor de origem de dados está utilizando as mesmas configurações de modo de arredondamento DECFLOAT. Os valores válidos são Y e N. O padrão é N; o servidor federado e o servidor remoto possuem configurações do modo de arredondamento DECFLOAT diferentes.</p> <p>Importante: Se você configurar esta opção como Y quando os modos de arredondamento são diferentes entre o servidor federado e o servidor de origem de dados, é possível receber os resultados de arredondamento DECFLOAT incorretos.</p> <p>Para configurar um servidor federado existente e servidor de origem de dados para utilizarem a mesma configuração do modo de arredondamento DECFLOAT, utilize a instrução ALTER SERVER.</p> <p>Restrição: Esta opção do servidor é válida apenas para o DB2 Database para Linux, UNIX e Windows Versão 9.5 e posterior.</p> <p>In InfoSphere Federation Server Version 9.7 Fix Pack 2 and later, when you run the CREATE SERVER statement, this server option is automatically configured based on the configuration of your data source. If you attempt to manually configure this server option, you receive the SQL1841N message.</p>

Tabela 52. Opções de Servidor para as Origens de Dados do DB2 (continuação)

Nome	Descrição
VARCHAR2_COMPAT	<p>Especifica se a origem de dados remota é compatível com VARCHAR2. Os valores válidos são Y e N. O valor-padrão varia dependendo da origem de dados remota. Para origens de dados remotas Oracle, o padrão é Y; a origem de dados é compatível com VARCHAR2. Para todas as outras origens de dados remotas, o padrão é N; a origem de dados não é configurada no modo compatível com VARCHAR2.</p> <p>É necessário configurar esta opção do servidor como Y, se a origem de dados DB2 Database para Linux, UNIX e Windows, ODBC ou JDBC estiver configurada no modo compatível com VARCHAR2.</p> <p>In InfoSphere Federation Server Version 9.7 Fix Pack 2 and later, when you run the CREATE SERVER statement, this server option is automatically configured based on the configuration of your data source. If you attempt to manually configure this server option, you receive the SQL1841N message.</p>

Opções de Mapeamento de Usuários

Tabela 53. Opções de Mapeamento de Usuários para Origens de Dados do DB2

Opção	Descrição
FED_PROXY_USER	<p>Especifica o ID de autorização utilizado para estabelecer todas as conexões confiáveis de saída, quando a conexão de entrada não é confiável. O usuário cujo ID é especificado nesta opção deve ter um mapeamento de usuário que especifica REMOTE_AUTHID e REMOTE_PASSWORD. Se você especificar a opção de mapeamento de usuário FED_PROXY_USER, é necessário também especificar a opção do servidor FED_PROXY_USER.</p> <p>Restrição: This server option is only valid for DB2 Database para Linux, UNIX e Windows Version 9.5 and later and DB2 for z/OS Version 9 and later.</p>
ACCOUNTING_STRING	<p>Necessário se os dados da conta devem ser passados. Especifica uma cadeia de contabilidade DRDA. Os valores válidos incluem qualquer cadeia com 255 caracteres ou menos.</p>
REMOTE_AUTHID	<p>Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.</p>

Tabela 53. Opções de Mapeamento de Usuários para Origens de Dados do DB2 (continuação)

Opção	Descrição
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.
USE_TRUSTED_CONTEXT	Especifica se o mapeamento de usuário é confiável. Os valores válidos são Y e N. O padrão é N; o mapeamento de usuário não é confiável e pode ser utilizado apenas em conexões de saída federadas não confiáveis. Y especifica que o mapeamento de usuário é confiável e pode ser utilizado em conexões de saída federadas confiáveis e não confiáveis. Restrição: This server option is only valid for DB2 Database para Linux, UNIX e Windows Version 9.5 and later and DB2 for z/OS Version 9 and later.

Opções de Coluna

Tabela 54. Opções de Colunas para as Origens de Dados do DB2

Opção	Descrição
NUMERIC_STRING	Especifica como tratar as cadeias numéricas. O padrão é N. Se a coluna de cadeias da origem de dados contiver apenas cadeias numéricas e nenhum outro caractere, incluindo caracteres em branco, configure a opção NUMERIC_STRING para Y. Quando NUMERIC_STRING é configurado como Y para uma coluna, o otimizador de consultas reconhece que a coluna não contém caracteres em branco que poderiam interferir com a classificação dos dados na coluna. Utilize esta opção quando a sequência de intercalação de uma origem de dados for diferente da sequência de intercalação que o servidor federado utiliza. As colunas que utilizam esta opção não são excluídas da avaliação remota, devido a uma sequência de intercalação diferente.
NO_EMPTY_STRING	Especifica se o servidor de origem de dados remoto pode conter cadeias vazias. Os valores válidos são Y e N. O valor-padrão varia dependendo da origem de dados remota. Para as origens de dados remotas do Oracle, o padrão é Y; todos os valores de cadeias vazias são convertidos em valores NULL. Para todas as outras origens de dados remotas, o padrão é N; a origem de dados pode conter cadeias vazias.

Tabela 54. Opções de Colunas para as Origens de Dados do DB2 (continuação)

Opção	Descrição
XML_ROOT	Especifica o elemento-raiz XML a ser incluído nos valores de uma coluna XML que referencia uma sequência XML. This option ensures that the values of the XML column are a well-formed XML document.

Referência de Opções do Excel

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique as opções de wrapper, servidor e apelido.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 55. Opções de Wrapper para Excel

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.

Opções do Servidor

Tabela 56. Opções de Servidor para Excel

Nome	Descrição
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.

Opções de Apelido

Tabela 57. Opções de Apelido para Excel

Opção	Descrição
FILE_PATH	Necessário. Especifica o caminho do diretório completo e o nome do arquivo da planilha Excel que deseja acessar.
RANGE	Especifica o intervalo das células a serem utilizadas; por exemplo, A1:C100. O valor antes da vírgula especifica a célula superior esquerda do intervalo. O valor após da vírgula especifica a célula inferior direita do intervalo.

Referência das Opções do Informix

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, o servidor, o mapeamento de usuário e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 58. Opções de Wrapper para Informix

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Opções do Servidor

Tabela 59. Opções de Servidor para Informix

Nome	Descrição
COLLATING_SEQUENCE	Especifica se a origem de dados especifica a mesma seqüência de intercalação padrão que o banco de dados federado. Os valores válidos são Y, N e I. I especifica a não distinção entre maiúsculas e minúsculas. O padrão é Y. A seqüência de intercalação especificada para o o servidor federado deve corresponder à seqüência de intercalação na origem de dados remota.
COMM_RATE	Especifica a taxa de comunicação, em megabytes por segundo, entre o servidor federado e o servidor de origem de dados. Os valores válidos são números inteiros maiores que 0 e menores que 2147483648. O padrão é 2.

Tabela 59. Opções de Servidor para Informix (continuação)

Nome	Descrição
CPU_RATIO	Especifica o quão mais rápida ou mais lenta a CPU de origem de dados é em comparação à CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de CPU; uma proporção 1:1. Uma configuração 0.5 indica que a CPU do servidor federado é 50% mais lenta que a CPU da origem de dados. Uma configuração 2 indica que a CPU federada é duas vezes mais rápida que a CPU da origem de dados.
DBNAME	Necessário. Especifica o nome do banco de dados Informix que você deseja acessar.
DB2_MAXIMAL_PUSHDOWN	Especifica os critérios primários que o otimizador de consultas utiliza para escolher um plano de acesso. Os valores válidos são Y e N. O padrão é N; o otimizador de consultas escolhe o plano que possui o menor custo estimado. Y especifica que o otimizador de consultas escolhe o plano de acesso que direciona a maioria das operações de consulta à origem de dados. Se mais de um plano de acesso corresponder aos critérios, o plano que possui o menor custo é escolhido.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
DB2_TWO_PHASE_COMMIT	Especifica se o servidor federado se conecta à origem de dados no protocolo two-phase commit ou no protocolo one-phase commit. Os valores válidos são Y e N. O padrão é N; o servidor federado utiliza o protocolo one-phase commit para se conectar. Y especifica que o servidor federado utiliza o protocolo two-phase commit para se conectar.

Tabela 59. Opções de Servidor para Informix (continuação)

Nome	Descrição
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.
FOLD_ID	Especifica o caso para o ID do usuário que é enviado à origem de dados. Não há valor-padrão; o servidor federado envia o ID do usuário em maiúsculas; em seguida, se o ID do usuário em maiúsculas falhar, o servidor envia o ID do usuário em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.
FOLD_PW	Especifica o caso para a senha que é enviada à origem de dados. Não há valor-padrão; o servidor federado envia a senha em maiúsculas; em seguida, se a senha em maiúsculas falhar, o servidor envia a senha em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.
INFORMIX_CLIENT_LOCALE	Especifica o CLIENT_LOCALE a ser utilizado para a conexão entre o servidor federado e o servidor de origem de dados. O valor é qualquer código do idioma válido do Informix. Se esta opção não for especificada, a variável de ambiente CLIENT_LOCALE é configurada com o valor que é especificado no arquivo db2dj.ini. Se db2dj.ini não especificar a variável de ambiente CLIENT_LOCALE, INFORMIX_CLIENT_LOCALE é configurado como código do idioma do Informix, que melhor corresponde à página de códigos e território do banco de dados federado.

Tabela 59. Opções de Servidor para Informix (continuação)

Nome	Descrição
INFORMIX_DB_LOCALE	Especifica o Informix DB_LOCALE a ser utilizado para a conexão entre o servidor federado e o servidor de origem de dados. Se a opção INFORMIX_DB_LOCALE não for especificada, a variável de ambiente Informix DB_LOCALE é configurada com o valor especificado no arquivo db2dj.ini. Se o arquivo db2dj.ini não especificar um valor, a variável de ambiente Informix DB_LOCALE não é configurada.
INFORMIX_LOCK_MODE	Especifica o modo de bloqueio configurado para uma origem de dados do Informix. O wrapper do Informix emite o comando SET LOCK MODE imediatamente após conectar-se a uma origem de dados do Informix. Os valores válidos são W, N e um número. O padrão é W; o wrapper aguarda um período de tempo ilimitado para que o bloqueio seja liberado. N especifica que não deve aguardar; um erro é retornado imediatamente. Utilize um número para especificar o período máximo de tempo, em segundos, a aguardar. Se um conflito ou tempo limite ocorrer, utilize a instrução ALTER SERVER para alterar o valor da opção INFORMIX_LOCK_MODE. Por Exemplo: ALTER SERVER TYPE informix VERSION 9 WRAPPER informix OPÇÕES (ADD informix_lock_mode '60')
IO_RATIO	Especifica o quão mais rápido ou mais lento o sistema I/O de origem de dados é executado em comparação ao sistema I/O do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de I/O; uma proporção 1:1. Uma configuração 0.5 indica que a velocidade do servidor federado é 50% mais lenta que a velocidade da origem de dados. Uma configuração 2 indica que a velocidade federada é duas vezes mais rápida que a velocidade da origem de dados.

Tabela 59. Opções de Servidor para Informix (continuação)

Nome	Descrição
IUD_APP_SVPT_ENFORCE	Especifica se o servidor federado impinge o uso das instruções de ponto de salvamento do aplicativo. Os valores válidos são Y e N. O padrão é Y; se a origem de dados não impingir as instruções do ponto de salvamento do aplicativo e um erro ocorrer durante uma operação de inserção, atualização ou exclusão, o servidor federado atrasa a transação e o código de erro SQL SQL1476N é retornado. Recomenda-se utilizar a configuração padrão.
NODE	Necessário. Especifica o nome através do qual a origem de dados é definida como uma instância para esse sistema de gerenciamento de banco de dados relacional.
OLD_NAME_GEN	Especifica como converter os nomes das colunas e os nomes de índices que estão na origem de dados nos nomes das colunas de apelidos e nomes de índices locais para o servidor federado. Os valores válidos são Y e N. O padrão é N; os nomes gerados correspondem fortemente aos nomes na origem de dados. Y especifica que os nomes gerados são os mesmos nomes criados no IBM WebSphere Federation Server Versão 9 e anterior. Assim, os nomes podem não corresponder tão fortemente aos nomes da origem de dados.
PUSHDOWN	Especifica se o servidor federado permite que a origem de dados avalie as operações. Os valores válidos são Y e N. O padrão é Y; a origem de dados avalia as operações. N especifica que o servidor federado envia instruções SQL que incluem apenas SELECT com os nomes das colunas. Predicados, como a coluna WHERE=;, as funções escalares, como MAX e MIN;; as classificações, como ORDER BY OR GROUP BY;; e as junções não são incluídas no SQL que o servidor federado envia à origem de dados.

Opções de Mapeamento de Usuários

Tabela 60. Opções de Mapeamento de Usuários para Informix

Nome	Descrição
REMOTE_AUTHID	Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.

Tabela 60. Opções de Mapeamento de Usuários para Informix (continuação)

Nome	Descrição
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.

Opções de Coluna

Tabela 61. Opções de Colunas para Informix

Nome	Descrição
BINARY_REP	Identifies specific binary data type columns. You can set the value of the BINARY_REP column option to Y to force the federated server to push down the SQL_BINARY data type. Enabling the query optimizer to perform the comparison of binary type columns at the remote data source can improve query performance.
NUMERIC_STRING	Especifica como tratar as cadeias numéricas. O padrão é N. Se a coluna de cadeias da origem de dados contiver apenas cadeias numéricas e nenhum outro caractere, incluindo caracteres em branco, configure a opção NUMERIC_STRING para Y. Quando NUMERIC_STRING é configurado como Y para uma coluna, o otimizador de consultas reconhece que a coluna não contém caracteres em branco que poderiam interferir com a classificação dos dados na coluna. Utilize esta opção quando a sequência de intercalação de uma origem de dados for diferente da sequência de intercalação que o servidor federado utiliza. As colunas que utilizam esta opção não são excluídas da avaliação remota, devido a uma sequência de intercalação diferente.

Referência de Opções JDBC

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, o servidor, o mapeamento de usuário e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 62. Opções de Wrapper para JDBC

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. O único valor válido é Y, por que o servidor DB2 suporta apenas o carregamento de JVM no modo em fence. O padrão é Y; o wrapper é executado no modo em fence.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Opções do Servidor

Tabela 63. Opções de Servidor para JDBC

Nome	Descrição
COLLATING_SEQUENCE	Especifica se a origem de dados especifica a mesma seqüência de intercalação padrão que o banco de dados federado. Os valores válidos são Y, N e I. I especifica a não distinção entre maiúsculas e minúsculas. O padrão é Y. A seqüência de intercalação especificada para o o servidor federado deve corresponder à seqüência de intercalação na origem de dados remota.
COMM_RATE	Especifica a taxa de comunicação, em megabytes por segundo, entre o servidor federado e o servidor de origem de dados. Os valores válidos são números inteiros maiores que 0 e menores que 2147483648. O padrão é 2.

Tabela 63. Opções de Servidor para JDBC (continuação)

Nome	Descrição
CPU_RATIO	Especifica o quão mais rápida ou mais lenta a CPU de origem de dados é em comparação à CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de CPU; uma proporção 1:1. Uma configuração 0.5 indica que a CPU do servidor federado é 50% mais lenta que a CPU da origem de dados. Uma configuração 2 indica que a CPU federada é duas vezes mais rápida que a CPU da origem de dados.
DATEFORMAT	Especifica o formato de data que a origem de dados utiliza. Utilize 'DD', 'MM' e 'YY' ou 'YYYY' para especificar o formato de data. É possível especificar um delimitador, como um espaço, hífen ou vírgula. Por exemplo, o formato 'YYYY-MM-DD' especifica uma data como 1958-10-01. O valor pode conter valores nulos.
DB2_MAXIMAL_PUSHDOWN	Especifica os critérios primários que o otimizador de consultas utiliza para escolher um plano de acesso. Os valores válidos são Y e N. O padrão é N; o otimizador de consultas escolhe o plano que possui o menor custo estimado. Y especifica que o otimizador de consultas escolhe o plano de acesso que direciona a maioria das operações de consulta à origem de dados.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 0. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Tabela 63. Opções de Servidor para JDBC (continuação)

Nome	Descrição
DRIVER_CLASS	<p>Especifica a biblioteca de drivers JDBC. É possível registrar o servidor novamente em relação a várias origens de dados do JDBC, se o driver JDBC estiver em conformidade com a especificação JDBC versão 3.0 e posterior. Consulte a documentação do driver JDBC para obter as especificações e informações JDBC sobre como configurar a opção do servidor DRIVER_CLASS.</p> <p>Exemplo</p> <p>Neste exemplo, a biblioteca de drivers JDBC com <code>com.ibm.db2.jcc.DB2Driver</code> é especificada:</p> <pre>DRIVER_CLASS 'com.ibm.db2.jcc.DB2Driver'</pre> <p>Importante: Se você especificar esta opção, é necessário também especificar a opção do servidor URL.</p>
DRIVER_PACKAGE	<p>Especifica os pacotes de driver JDBC. Utilize um separador de caminho para especificar vários pacotes de classe de drivers. Utilize um ponto e vírgula nos sistemas operacionais Windows e uma vírgula nos sistemas operacionais Linux e Unix.</p> <p>Exemplo</p> <p>Neste exemplo, você especifica vários pacotes de drivers com uma vírgula nos sistemas operacionais Linux:</p> <pre>DRIVER_PACKAGE '/path1/file1.jar: /path2/file2.jar'</pre>
FOLD_ID	<p>Especifica o caso para o ID do usuário que é enviado à origem de dados. Não há valor-padrão; o servidor federado envia o ID do usuário em maiúsculas; em seguida, se o ID do usuário em maiúsculas falhar, o servidor envia o ID do usuário em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.</p>
FOLD_PW	<p>Especifica o caso para a senha que é enviada à origem de dados. Não há valor-padrão; o servidor federado envia a senha em maiúsculas; em seguida, se a senha em maiúsculas falhar, o servidor envia a senha em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.</p>

Tabela 63. Opções de Servidor para JDBC (continuação)

Nome	Descrição
IO_RATIO	<p>Especifica o quão mais rápido ou mais lento o sistema I/O de origem de dados é executado em comparação ao sistema I/O do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23}. O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de I/O; uma proporção 1:1. Uma configuração 0.5 indica que a velocidade do servidor federado é 50% mais lenta que a velocidade da origem de dados. Uma configuração 2 indica que a velocidade federada é duas vezes mais rápida que a velocidade da origem de dados.</p>
JDBC_LOG	<p>Especifica se o wrapper JDBC cria arquivos de log para o rastreamento de erros. Os valores válidos são Y e N. O padrão é N; o arquivo de log não é criado. Se essa opção do servidor for configurada como Y, o wrapper JDBC grava os arquivos de log JDBC no arquivo <code>jdbc_wrapper_prod_id.log</code>, em que <i>prod_id</i> é o ID do produto. O arquivo de log é armazenado no diretório especificado pelo parâmetro de configuração do gerenciador de bancos de dados DB2, DIAGPATH. O diretório padrão nos sistemas UNIX é <code>inst_home/sql11ib/db2dump</code>.</p> <p>Recomendação: A configuração desta opção do servidor como YES terá um impacto no desempenho do sistema e você não deve ativar o log em sistemas de produção.</p>
OLD_NAME_GEN	<p>Especifica como converter os nomes das colunas e os nomes de índices que estão na origem de dados nos nomes das colunas de apelidos e nomes de índices locais para o servidor federado. Os valores válidos são Y e N. O padrão é N; os nomes gerados correspondem fortemente aos nomes na origem de dados. Y especifica que os nomes gerados são os mesmos nomes criados no IBM WebSphere Federation Server Versão 9 e anterior. Assim, os nomes podem não corresponder tão fortemente aos nomes da origem de dados.</p>

Tabela 63. Opções de Servidor para JDBC (continuação)

Nome	Descrição
PUSHDOWN	Especifica se o servidor federado permite que a origem de dados avalie as operações. Os valores válidos são Y e N. O padrão é Y; a origem de dados avalia as operações. N especifica que o servidor federado envia instruções SQL que incluem apenas SELECT com os nomes das colunas. Predicados, como a coluna WHERE=;, as funções escalares, como MAX e MIN;,, as classificações, como ORDER BY OR GROUP BY;,, e as junções não são incluídas no SQL que o servidor federado envia à origem de dados.
TIMEFORMAT	Especifica o formato de hora que a origem de dados utiliza. Utilize 'hh12', 'hh24', 'mm', 'ss', 'AM' e 'A.M.' para especificar o formato de hora. Por exemplo, o formato 'hh24:mm:22' especifica uma hora como 16:00:00. O formato 'hh12:mm:ss AM' especifica uma hora como 8:00:00 AM. O valor pode conter valores nulos.
TIMESTAMPFORMAT	Especifica o formato de registro de data e hora que a origem de dados utiliza. Os valores válidos estão no formato que a opção DATEFORMAT e a opção TIMEFORMAT utilizam. Especifique 'n' para um décimo de segundo, 'nn' para um centésimo de segundo, 'nnn' para milissegundos, etc., até 'nnnnnn' para microssegundos. Por exemplo, o formato 'YYY-MM-DD-hh24:mm:ss.nnnnnn' especifica um registro de data e hora como 1994-01-01-24:00:00.000000. O valor pode conter valores nulos.

Tabela 63. Opções de Servidor para JDBC (continuação)

Nome	Descrição
URL	<p>Especifica a cadeia de conexão JDBC do servidor remoto.</p> <p>A cadeia de conexão JDBC consiste de três partes, separadas por uma vírgula:</p> <ul style="list-style-type: none"> • Protocolo de banco de dados • Nome do tipo de banco de dados ou nome do driver de conectividade • Identidade do banco de dados através de um alias ou sub-nome <p>Exemplo</p> <p>Neste exemplo, a cadeia de conexão JDBC é jdbc:db2://cn.ibm.com:50471/testdb:</p> <p>URL 'jdbc:db2://cn.ibm.com:50471/testdb'</p> <p>É possível registrar o servidor em relação a várias origens de dados do JDBC, se o driver JDBC estiver em conformidade com a especificação JDBC versão 3.0 e superior. Consulte a documentação do driver JDBC para obter as especificações JDBC e informações sobre como configurar a opção do servidor URL.</p> <p>Importante: Se você especificar esta opção, é necessário também especificar a opção do servidor DRIVER_CLASS.</p>
VARCHAR_NO_TRAILING_BLANKS	<p>Especifica se a origem de dados contém colunas VARCHAR que contém pelo menos um caractere de rastreamento de espaço em branco. O padrão é N; as colunas VARCHAR contém pelo menos um caractere de rastreamento de espaço em branco.</p>

Opções de Mapeamento de Usuários

Tabela 64. Opções de Mapeamento de Usuários para JDBC

Nome	Descrição
REMOTE_AUTHID	<p>Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.</p>
REMOTE_PASSWORD	<p>Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.</p>

Opções de Coluna

Tabela 65. Opções de Colunas para JDBC

Nome	Descrição
NUMERIC_STRING	Especifica se a coluna contém cadeias de caracteres numéricos que incluem caracteres em branco. Os valores válidos são Y e N. O padrão é N; a coluna não contém cadeias numéricas que incluem caracteres em branco. Se a coluna contiver apenas cadeias numéricas seguidas por rastreamento de espaço em branco, não especifique Y. Quando NUMERIC_STRING é configurado como Y para uma coluna, o otimizador de consultas reconhece que a coluna não contém caracteres em branco que poderiam interferir com a classificação dos dados na coluna. Utilize esta opção quando a sequência de intercalação de uma origem de dados for diferente da sequência de intercalação que o servidor federado utiliza. As colunas que utilizam esta opção não são excluídas da avaliação remota, devido a uma sequência de intercalação diferente.
VARCHAR_NO_TRAILING_BLANKS	Especifica se há pelo menos um rastreamento de espaço em branco na coluna VARCHAR.

Referência de Opções do Microsoft SQL Server

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, o servidor, o mapeamento de usuário e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 66. Opções de Wrapper para Microsoft SQL Server

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.

Tabela 66. Opções de Wrapper para Microsoft SQL Server (continuação)

Nome	Descrição
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Opções do Servidor

Tabela 67. Opções de Servidor para Microsoft SQL Server

Nome	Descrição
CODEPAGE	Especifica a página de códigos que corresponde ao conjunto de caracteres codificado da configuração do cliente de origem de dados. Nos sistemas UNIX e Microsoft Windows que utilizam um banco de dados federado não-Unicode, o padrão é a página de códigos do banco de dados federado. Nos sistemas UNIX que utilizam um banco de dados federado do Unicode, o padrão é 1208. Nos sistemas Windows que utilizam um banco de dados federado do Unicode, o padrão é 1202.
COLLATING_SEQUENCE	Especifica se a origem de dados especifica a mesma seqüência de intercalação padrão que o banco de dados federado. Os valores válidos são Y, N e I. I especifica a não distinção entre maiúsculas e minúsculas. O padrão é Y. A seqüência de intercalação especificada para o o servidor federado deve corresponder à seqüência de intercalação na origem de dados remota.
COMM_RATE	Especifica o quão mais rápida ou mais lenta a CPU de origem de dados é em comparação à CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de CPU; uma proporção 1:1. Uma configuração 0.5 indica que a CPU do servidor federado é 50% mais lenta que a CPU da origem de dados. Uma configuração 2 indica que a CPU federada é duas vezes mais rápida que a CPU da origem de dados.
CPU_RATIO	Especifica o quão mais rápido ou mais lento a CPU de origem de dados é executada em comparação à velocidade da CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4.

Tabela 67. Opções de Servidor para Microsoft SQL Server (continuação)

Nome	Descrição
DBNAME	Necessário. Especifica o alias para o banco de dados que você deseja acessar. O valor faz distinção entre maiúsculas e minúsculas.
DB2_MAXIMAL_PUSHDOWN	Especifica os critérios primários que o otimizador de consultas utiliza para escolher um plano de acesso. Os valores válidos são Y e N. O padrão é N; o otimizador de consultas escolhe o plano que possui o menor custo estimado. Y especifica que o otimizador de consultas escolhe o plano de acesso que direciona a maioria das operações de consulta à origem de dados. Se mais de um plano de acesso corresponder aos critérios, o plano que possui o menor custo é escolhido.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
DB2_TWO_PHASE_COMMIT	Especifica se o servidor federado se conecta à origem de dados no protocolo two-phase commit ou no protocolo one-phase commit. Os valores válidos são Y e N. O padrão é N; o servidor federado utiliza o protocolo one-phase commit para se conectar. Y especifica que o servidor federado utiliza o protocolo two-phase commit para se conectar. Importante: Se configurar esta opção como Y, é necessário especificar também XA_OPEN_STRING_OPTION.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Tabela 67. Opções de Servidor para Microsoft SQL Server (continuação)

Nome	Descrição
FOLD_ID	Especifica o caso para o ID do usuário que é enviado à origem de dados. Não há valor-padrão; o servidor federado envia o ID do usuário em maiúsculas; em seguida, se o ID do usuário em maiúsculas falhar, o servidor envia o ID do usuário em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.
FOLD_PW	Especifica o caso para a senha que é enviada à origem de dados. Não há valor-padrão; o servidor federado envia a senha em maiúsculas; em seguida, se a senha em maiúsculas falhar, o servidor envia a senha em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.
IO_RATIO	Especifica o quão mais rápido ou mais lento o sistema I/O de origem de dados é executado em comparação ao sistema I/O do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de I/O; uma proporção 1:1. Uma configuração 0.5 indica que a velocidade do servidor federado é 50% mais lenta que a velocidade da origem de dados. Uma configuração 2 indica que a velocidade federada é duas vezes mais rápida que a velocidade da origem de dados.
NODE	Necessário. Se o servidor federado utilizar Microsoft Windows, o valor de NODE é o nome DSN do sistema, especificado para o Microsoft SQL Server. Se o servidor federado utilizar UNIX or Linux, o valor de NODE é definido no arquivo <code>odbc.ini</code> . O valor faz distinção entre maiúsculas e minúsculas.

Tabela 67. Opções de Servidor para Microsoft SQL Server (continuação)

Nome	Descrição
OLD_NAME_GEN	Especifica como converter os nomes das colunas e os nomes de índices que estão na origem de dados nos nomes das colunas de apelidos e nomes de índices locais para o servidor federado. Os valores válidos são Y e N. O padrão é N; os nomes gerados correspondem fortemente aos nomes na origem de dados. Y especifica que os nomes gerados são os mesmos nomes criados no IBM WebSphere Federation Server Versão 9 e anterior. Assim, os nomes podem não corresponder tão fortemente aos nomes da origem de dados.
PASSWORD	Especifica se as senhas são ou não enviadas a uma origem de dados. O padrão é Y.
PUSHDOWN	Especifica se o servidor federado permite que a origem de dados avalie as operações. Os valores válidos são Y e N. O padrão é Y; a origem de dados avalia as operações. N especifica que o servidor federado envia instruções SQL que incluem apenas SELECT com os nomes das colunas. Predicados, como a coluna WHERE=;, as funções escalares, como MAX e MIN; as classificações, como ORDER BY OR GROUP BY; e as junções não são incluídas no SQL que o servidor federado envia à origem de dados.
XA_OPEN_STRING_OPTIONS	Necessário quando DB2_TWO_PHASE_COMMIT é configurado como Y. Especifica o ID do gerenciador de recursos do Microsoft SQL Server Registry.

Opções de Mapeamento de Usuários

Tabela 68. Opções de Mapeamento de Usuários para Microsoft SQL Server

Nome	Descrição
REMOTE_AUTHID	Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.

Opções de Coluna

Tabela 69. Opções de Coluna para Microsoft SQL Server

Nome	Descrição
BINARY_REP	Identifies specific binary data type columns. You can set the value of the BINARY_REP column option to Y to force the federated server to push down the SQL_BINARY data type. Enabling the query optimizer to perform the comparison of binary type columns at the remote data source can improve query performance.
NUMERIC_STRING	Especifica como tratar as cadeias numéricas. O padrão é N. Se a coluna de cadeias da origem de dados contiver apenas cadeias numéricas e nenhum outro caractere, incluindo caracteres em branco, configure a opção NUMERIC_STRING para Y. Quando NUMERIC_STRING é configurado como Y para uma coluna, o otimizador de consultas reconhece que a coluna não contém caracteres em branco que poderiam interferir com a classificação dos dados na coluna. Utilize esta opção quando a sequência de intercalação de uma origem de dados for diferente da sequência de intercalação que o servidor federado utiliza. As colunas que utilizam esta opção não são excluídas da avaliação remota, devido a uma sequência de intercalação diferente.

Referência de Opções ODBC

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, o servidor, o mapeamento de usuário e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 70. Opções de Wrapper para ODBC

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável. Importante: Se você configurar esta opção como Y para um sistema UNIX, é necessário também configurar a opção de wrapper DB2_SOURCE_CLIENT_MODE.

Tabela 70. Opções de Wrapper para ODBC (continuação)

Nome	Descrição
DB2_SOURCE_CLIENT_MODE	Especifica que o cliente para a origem de dados é de 32 bits e a instância de banco de dados no servidor federado é de 64 bits. O único valor válido é 32 bits. Esta opção é válida apenas para UNIX. Importante: Se você configurar esta opção, é necessário também configurar a opção de wrapper DB2_FENCED como Y.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.
MODULE	Necessário para servidores federados, executados em um sistema UNIX. Especifica o caminho completo da biblioteca que contém a implementação do ODBC Driver Manager ou a implementação do SQL/CLI. Não há padrão para UNIX. Em um sistema Microsoft Windows, o padrão é odbc32.dll.

Opções do Servidor

Tabela 71. Opções de Servidor para ODBC

Nome	Descrição
CODEPAGE	Especifica a página de códigos que corresponde ao conjunto de caracteres codificado da configuração do cliente para a origem de dados. Nos sistemas UNIX e Windows que utilizam um banco de dados federado não-Unicode, o padrão é a página de códigos que o banco de dados federado utiliza. Nos sistemas UNIX que utilizam um banco de dados federado do Unicode, o padrão é 1208. Nos sistemas Windows que utilizam um banco de dados federado do Unicode, o padrão é 1202.
COLLATING_SEQUENCE	Especifica se a origem de dados especifica a mesma sequência de intercalação padrão que o banco de dados federado. Os valores válidos são Y, N e I. I especifica a não distinção entre maiúsculas e minúsculas. O padrão é Y. A sequência de intercalação especificada para o o servidor federado deve corresponder à sequência de intercalação na origem de dados remota.

Tabela 71. Opções de Servidor para ODBC (continuação)

Nome	Descrição
COMM_RATE	Especifica a taxa de comunicação, em megabytes por segundo, entre o servidor federado e o servidor de origem de dados. Os valores válidos são números inteiros maiores que 0 e menores que 2147483648. O padrão é 2.
CPU_RATIO	Especifica o quão mais rápida ou mais lenta a CPU de origem de dados é em comparação à CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de CPU; uma proporção 1:1. Uma configuração 0.5 indica que a CPU do servidor federado é 50% mais lenta que a CPU da origem de dados. Uma configuração 2 indica que a CPU federada é duas vezes mais rápida que a CPU da origem de dados.
DATEFORMAT	Especifica o formato de data que a origem de dados utiliza. Utilize 'DD', 'MM' e 'YY' ou 'YYYY' para especificar o formato de data. É possível especificar um delimitador, como um espaço, hífen ou vírgula. Por exemplo, o formato 'YYYY-MM-DD' especifica uma data como 1958-10-01. O valor pode conter valores nulos. If you specify a value with single quotes, you need to preserve the single quotes. Specify additional single quotes, ' ', to preserve the single quotes. These are two additional single quotes, not a double quote. For example: ' ' 'YYYY-MM-DD' ' '.
DBNAME	Especifica o nome do banco de dados de origem de dados que deseja acessar.
DB2_AUTHID_QUOTE_CHAR	Specifies the quote characters that are used for authid names such as schema and user names. If you do not specify this option, double quotes are used by default.
DB2_ID_QUOTE_CHAR	Specifies the quote characters that are used for delimited identifiers such as column names. If you do not specify this option, double quotes are used by default.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 0. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.

Tabela 71. Opções de Servidor para ODBC (continuação)

Nome	Descrição
DB2_MAXIMAL_PUSHDOWN	Especifica os critérios primários que o otimizador de consultas utiliza para escolher um plano de acesso. Os valores válidos são Y e N. O padrão é N; o otimizador de consultas escolhe o plano que possui o menor custo estimado. Y especifica que o otimizador de consultas escolhe o plano de acesso que direciona a maioria das operações de consulta à origem de dados.
DB2_TABLE_QUOTE_CHAR	Specifies the quote characters that are used for table names. If you do not specify this option, double quotes are used by default.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.
ENABLE_BULK_INSERT	Specifies whether bulk insert processing is enabled for the Netezza data source. Os valores válidos são Y e N. O padrão é N.
FOLD_ID	Especifica o caso para o ID do usuário que é enviado à origem de dados. Não há valor-padrão; o servidor federado envia o ID do usuário em maiúsculas; em seguida, se o ID do usuário em maiúsculas falhar, o servidor envia o ID do usuário em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.
FOLD_PW	Especifica o caso para a senha que é enviada à origem de dados. Não há valor-padrão; o servidor federado envia a senha em maiúsculas; em seguida, se a senha em maiúsculas falhar, o servidor envia a senha em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.

Tabela 71. Opções de Servidor para ODBC (continuação)

Nome	Descrição
IO_RATIO	Especifica o quão mais rápido ou mais lento o sistema I/O de origem de dados é executado em comparação ao sistema I/O do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de I/O; uma proporção 1:1. Uma configuração 0.5 indica que a velocidade do servidor federado é 50% mais lenta que a velocidade da origem de dados. Uma configuração 2 indica que a velocidade federada é duas vezes mais rápida que a velocidade da origem de dados.
NODE	Necessário. Especifica o nome do nó ou o nome DNS do sistema designado à origem de dados ODBC, quando DSN é definido. O valor faz distinção entre maiúsculas e minúsculas.
OLD_NAME_GEN	Especifica como converter os nomes das colunas e os nomes de índices que estão na origem de dados nos nomes das colunas de apelidos e nomes de índices locais para o servidor federado. Os valores válidos são Y e N. O padrão é N; os nomes gerados correspondem fortemente aos nomes na origem de dados. Y especifica que os nomes gerados são os mesmos nomes criados no IBM WebSphere Federation Server Versão 9 e anterior. Assim, os nomes podem não corresponder tão fortemente aos nomes da origem de dados.
PUSHDOWN	Especifica se o servidor federado permite que a origem de dados avalie as operações. Os valores válidos são Y e N. O padrão é Y; a origem de dados avalia as operações. N especifica que o servidor federado envia instruções SQL que incluem apenas SELECT com os nomes das colunas. Predicados, como a coluna WHERE=;, as funções escalares, como MAX e MIN;; as classificações, como ORDER BY OR GROUP BY;; e as junções não são incluídas no SQL que o servidor federado envia à origem de dados.
TIMEFORMAT	Especifica o formato de hora que a origem de dados utiliza. Utilize 'hh12', 'hh24', 'mm', 'ss', 'AM' e 'A.M.' para especificar o formato de hora. Por exemplo, o formato 'hh24:mm:22' especifica uma hora como 16:00:00. O formato 'hh12:mm:ss AM' especifica uma hora como 8:00:00 AM. O valor pode conter valores nulos.

Tabela 71. Opções de Servidor para ODBC (continuação)

Nome	Descrição
TIMESTAMPFORMAT	Especifica o formato de registro de data e hora que a origem de dados utiliza. Os valores válidos estão no formato que a opção DATEFORMAT e a opção TIMEFORMAT utilizam. Especifique 'n' para um décimo de segundo, 'nn' para um centésimo de segundo, 'nnn' para milissegundos, etc., até 'nnnnnn' para microssegundos. Por exemplo, o formato 'YYY-MM-DD-hh24:mm:ss.nnnnnn' especifica um registro de data e hora como 1994-01-01-24:00:00.000000. O valor pode conter valores nulos.
VARCHAR_NO_TRAILING_BLANKS	Especifica se a origem de dados contém colunas VARCHAR que contém pelo menos um caractere de rastreamento de espaço em branco. O padrão é N; as colunas VARCHAR contém pelo menos um caractere de rastreamento de espaço em branco.

Opções de Mapeamento de Usuários

Tabela 72. Opções de Mapeamento de Usuários para ODBC

Nome	Descrição
REMOTE_AUTHID	Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.

Opções de Coluna

Tabela 73. Opções de Colunas para ODBC

Nome	Descrição
NUMERIC_STRING	Especifica se a coluna contém cadeias de caracteres numéricos que incluem caracteres em branco. Os valores válidos são Y e N. O padrão é N; a coluna não contém cadeias numéricas que incluem caracteres em branco. Se a coluna contiver apenas cadeias numéricas seguidas por rastreamento de espaço em branco, não especifique Y. Quando NUMERIC_STRING é configurado como Y para uma coluna, o otimizador de consultas reconhece que a coluna não contém caracteres em branco que poderiam interferir com a classificação dos dados na coluna. Utilize esta opção quando a sequência de intercalação de uma origem de dados for diferente da sequência de intercalação que o servidor federado utiliza. As colunas que utilizam esta opção não são excluídas da avaliação remota, devido a uma sequência de intercalação diferente.
VARCHAR_NO_TRAILING_BLANKS	Especifica se há pelo menos um rastreamento de espaço em branco na coluna VARCHAR.

Referência de Opções Oracle

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, o servidor, o mapeamento de usuário e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 74. Opções de Wrapper para Oracle

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.

Tabela 74. Opções de Wrapper para Oracle (continuação)

Nome	Descrição
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Opções do Servidor

Tabela 75. Opções de Servidor para Oracle

Nome	Descrição
COLLATING_SEQUENCE	Especifica se a origem de dados especifica a mesma seqüência de intercalação padrão que o banco de dados federado. Os valores válidos são Y, N e I. I especifica a não distinção entre maiúsculas e minúsculas. O padrão é Y. A seqüência de intercalação especificada para o o servidor federado deve corresponder à seqüência de intercalação na origem de dados remota.
COMM_RATE	Especifica a taxa de comunicação, em megabytes por segundo, entre o servidor federado e o servidor de origem de dados. Os valores válidos são números inteiros maiores que 0 e menores que 2147483648. O padrão é 2.
CPU_RATIO	Especifica o quão mais rápida ou mais lenta a CPU de origem de dados é em comparação à CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de CPU; uma proporção 1:1. Uma configuração 0.5 indica que a CPU do servidor federado é 50% mais lenta que a CPU da origem de dados. Uma configuração 2 indica que a CPU federada é duas vezes mais rápida que a CPU da origem de dados.
DB2_MAXIMAL_PUSHDOWN	Especifica os critérios primários que o otimizador de consultas utiliza para escolher um plano de acesso. Os valores válidos são Y e N. O padrão é N; o otimizador de consultas escolhe o plano que possui o menor custo estimado. Y especifica que o otimizador de consultas escolhe o plano de acesso que direciona a maioria das operações de consulta à origem de dados.

Tabela 75. Opções de Servidor para Oracle (continuação)

Nome	Descrição
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
DB2_TWO_PHASE_COMMIT	Especifica se o servidor federado se conecta à origem de dados no protocolo two-phase commit ou no protocolo one-phase commit. Os valores válidos são Y e N. O padrão é N; o servidor federado utiliza o protocolo one-phase commit para se conectar. Y especifica que o servidor federado utiliza o protocolo two-phase commit para se conectar.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.
FED_PROXY_USER	Especifica o ID de autorização utilizado para estabelecer todas as conexões confiáveis de saída, quando a conexão de entrada não é confiável. Importante: O usuário cujo ID é especificado nesta opção deve ter um mapeamento de usuário que especifica REMOTE_AUTHID e REMOTE_PASSWORD.
FOLD_ID	Especifica o caso para o ID do usuário que é enviado à origem de dados. Não há valor-padrão; o servidor federado envia o ID do usuário em maiúsculas; em seguida, se o ID do usuário em maiúsculas falhar, o servidor envia o ID do usuário em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.

Tabela 75. Opções de Servidor para Oracle (continuação)

Nome	Descrição
FOLD_PW	Especifica o caso para a senha que é enviada à origem de dados. Não há valor-padrão; o servidor federado envia a senha em maiúsculas; em seguida, se a senha em maiúsculas falhar, o servidor envia a senha em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.
IO_RATIO	Especifica o quão mais rápido ou mais lento o sistema I/O de origem de dados é executado em comparação ao sistema I/O do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de I/O; uma proporção 1:1. Uma configuração 0.5 indica que a velocidade do servidor federado é 50% mais lenta que a velocidade da origem de dados. Uma configuração 2 indica que a velocidade federada é duas vezes mais rápida que a velocidade da origem de dados.
NODE	Necessário. Especifica o nome do nó onde o servidor de banco de dados Oracle reside. Obtenha o nome do nó no arquivo <code>tnsnames.ora</code> .
OLD_NAME_GEN	Especifica como converter os nomes das colunas e os nomes de índices que estão na origem de dados nos nomes das colunas de apelidos e nomes de índices locais para o servidor federado. Os valores válidos são Y e N. O padrão é N; os nomes gerados correspondem fortemente aos nomes na origem de dados. Y especifica que os nomes gerados são os mesmos nomes criados no IBM WebSphere Federation Server Versão 9 e anterior. Assim, os nomes podem não corresponder tão fortemente aos nomes da origem de dados.

Tabela 75. Opções de Servidor para Oracle (continuação)

Nome	Descrição
PLAN_HINTS	Especifica se as sugestões de planos devem ou não ser ativadas. As sugestões de planos são fragmentos de instruções que fornecem informações adicionais que o otimizador de origem de dados utiliza para melhorar o desempenho das consultas. O otimizador de origem de dados utiliza as sugestões de planos para decidir se deve ou não utilizar um índice e qual índice ou qual sequência de junção de tabelas deve utilizar. Os valores válidos são Y e N. O padrão é N; as sugestões de planos não devem ser ativadas. Y especifica que as sugestões de planos devem ser ativadas na origem de dados, se suportar sugestões de planos.
PUSHDOWN	Especifica se o servidor federado permite que a origem de dados avalie as operações. Os valores válidos são Y e N. O padrão é Y; o servidor federado permite que a origem de dados avalie as operações. N especifica que o servidor federado recupera as colunas da origem de dados remota.
VARCHAR_NO_TRAILING_BLANKS	Especifica para um servidor particular, se as colunas VARCHAR contêm rastreios de espaço em branco. Para aplicar a opção a uma única coluna, utilize a opção de coluna VARCHAR_NO_TRAILING_BLANKS.
XA_OPEN_STRING_OPTIONS	Especifica informações adicionais para anexar à cadeia. Por exemplo, as informações podem ser o diretório para os arquivos de rastreio.

Opções de Mapeamento de Usuários

Tabela 76. Opções de Mapeamento de Usuários para Oracle

Nome	Descrição
FED_PROXY_USER	Especifica o ID de autorização utilizado para estabelecer todas as conexões confiáveis de saída, quando a conexão de entrada não é confiável. Importante: O usuário cujo ID é especificado nesta opção deve ter um mapeamento de usuário que especifica REMOTE_AUTHID e REMOTE_PASSWORD. Se você especificar a opção de mapeamento de usuário FED_PROXY_USER, é necessário também especificar a opção do servidor FED_PROXY_USER.
REMOTE_AUTHID	Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.

Tabela 76. Opções de Mapeamento de Usuários para Oracle (continuação)

Nome	Descrição
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.
USE_TRUSTED_CONTEXT	Especifica se o mapeamento de usuário é confiável. Os valores válidos são Y e N. O padrão é N; o mapeamento de usuário não é confiável e pode ser utilizado apenas em conexões de saída não confiáveis. Y especifica que o mapeamento de usuário é confiável e pode ser utilizado em conexões de saída confiáveis e não confiáveis.

Opções de Coluna

Tabela 77. Opções de Colunas para Oracle

Nome	Descrição
NUMERIC_STRING	Especifica se a coluna contém cadeias de caracteres numéricos que incluem caracteres em branco. Os valores válidos são Y e N. O padrão é N; a coluna não contém cadeias numéricas que incluem caracteres em branco. Se a coluna contiver apenas cadeias numéricas seguidas por rastreamento de espaço em branco, não especifique Y. Quando NUMERIC_STRING é configurado como Y para uma coluna, o otimizador de consultas reconhece que a coluna não contém caracteres em branco que poderiam interferir com a classificação dos dados na coluna. Utilize esta opção quando a sequência de intercalação de uma origem de dados for diferente da sequência de intercalação que o servidor federado utiliza. As colunas que utilizam esta opção não são excluídas da avaliação remota, devido a uma sequência de intercalação diferente.
VARCHAR_NO_TRAILING_BLANKS	Especifica se a coluna contém rastreamentos de espaço em branco.

Referência de Opções de Script

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, servidor, mapeamento de usuário, apelido e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 78. Opções de Wrapper para Scripts

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.
PROXY_TYPE	Especifica o tipo de proxy a ser utilizado para acessar a Internet, quando o servidor federado está protegido por firewall. Os valores válidos são NONE e SOCKS. O valor-padrão é NONE.
PROXY_SERVER_NAME	Especifica o nome ou endereço IP do servidor proxy. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.
PROXY_SERVER_PORT	Especifica a porta ou o nome do serviço para o serviço de proxy no servidor proxy. Os valores válidos são um número da porta decimal de 1 a 32760 ou um nome do serviço.

Opções do Servidor

Tabela 79. Opções de Servidor para Scripts

Nome	Descrição
DAEMON_PORT	Especifica o número de porta na qual o daemon do script atende aos pedidos de tarefas do script. O padrão é 4099. O número da porta deve ser o mesmo da opção DAEMON_PORT no arquivo de configuração do daemon. Se um nome do serviço for configurado para o daemon de script, utilize um nome de serviço TCP/IP.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
NODE	Necessário. Especifica o nome do host DNS ou endereço IP do sistema no qual o daemon de script é executado. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.
PROXY_AUTHID	Especifica o nome de usuário para a autenticação do servidor proxy.
PROXY_PASSWORD	Especifica a senha para a autenticação do servidor proxy.

Tabela 79. Opções de Servidor para Scripts (continuação)

Nome	Descrição
PROXY_SERVER_NAME	Especifica o nome ou endereço IP do servidor proxy. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.
PROXY_SERVER_PORT	Especifica a porta ou o nome do serviço para o serviço de proxy no servidor proxy. Os valores válidos são um número da porta decimal de 1 a 32760 ou um nome do serviço.
PROXY_TYPE	Especifica o tipo de proxy a ser utilizado para acessar a Internet, quando o servidor federado está protegido por firewall. Os valores válidos são NONE e SOCKS. O valor-padrão é NONE.

Opções de Mapeamento de Usuários

Tabela 80. Opções de Mapeamento de Usuários para Scripts

Nome	Descrição
PROXY_AUTHID	Especifica o nome de usuário para a autenticação do servidor proxy.
PROXY_PASSWORD	Especifica a senha para a autenticação do servidor proxy. A senha é criptografada quando é armazenada no catálogo de banco de dados federado.

Opções de Apelido

Tabela 81. Opções de Apelidos para Scripts

Nome	Descrição
DATASOURCE	Necessário para o apelido raiz. Especifica o nome do script a ser invocado. O script especificado como o valor desta opção também deve ser especificado no arquivo de configuração do daemon de script. Importante: Esta opção é válida apenas para o apelido raiz.
NAMESPACES	Especifica os espaços de nomes associados aos prefixos de espaço de nomes utilizados nas opções XPATH e TEMPLATE para cada coluna. Utilize esta sintaxe: <code>NAMESPACES 'prefix1="actual_namespace1", prefix2="actual_namespace2"'</code> Utilize uma vírgula para separar vários espaços de nomes. Por Exemplo: <code>NAMESPACES='http://www.myweb.com/cust', i='http://www.myweb.com/cust/id', n='http://www.myweb.com/cust/name'</code>

Tabela 81. Opções de Apelidos para Scripts (continuação)

Nome	Descrição
STREAMING	Especifica se o documento de origem deve ser separado em fragmentos lógicos para processamento. Os fragmentos correspondem ao nó que corresponde à expressão XPath do apelido. O wrapper então analisa e processa a origem de dados, fragmento por fragmento. Este tipo de análise minimiza o uso da memória. Os valores válidos são Y e N. O padrão é N; os documentos não são analisados. Esta opção é válida apenas em um apelido raiz. Não configure ambas as opções STREAMING e VALIDATE como Y.
TIMEOUT	Especifica o tempo máximo, em minutos, a aguardar pelos resultados do daemon. O padrão é 60. Esta opção é válida apenas para o apelido raiz.
VALIDATE	Especifica se o documento de origem é validado para assegurar sua conformidade com um esquema XML ou DTD (Document Type Definition) antes de extrair seus dados. O padrão é N; a validação não ocorre. Antes de configurar o valor como Y, o arquivo de esquema ou o arquivo DTD está no local que o documento de origem especifica. Esta opção é válida apenas para o apelido raiz. Não configure ambas as opções STREAMING e VALIDATE como Y.
XPATH	Especifica a expressão XPath que identifica os elementos XML, que representam tuplas individuais. A opção de apelido XPATH para um apelido filho é avaliada no contexto do caminho especificado pela opção de apelido XPath do pai. Esta expressão XPath é utilizada como um contexto para a avaliação dos valores de colunas que são identificados pela presença da opção de coluna XPath.

Opções de Coluna

Tabela 82. Opções de Colunas para Scripts

Nome	Descrição
DEFAULT	Especifica o valor-padrão para uma coluna de entrada de script. Se a consulta SQL não fornecer um valor, este valor-padrão é utilizado.

Tabela 82. Opções de Colunas para Scripts (continuação)

Nome	Descrição
FOREIGN_KEY	Indica que este apelido é um apelido filho e especifica o nome do apelido pai correspondente. Um apelido pode ter no máximo uma opção de coluna FOREIGN_KEY. O valor para a opção faz distinção entre maiúsculas e minúsculas. Não especifique a opção XPATH para esta coluna. A coluna pode ser utilizada apenas para juntar um apelido pai e um apelido filho. Uma instrução CREATE NICKNAME que inclui uma opção FOREIGN_KEY falha, se o apelido pai possuir um nome do esquema diferente. A menos que o apelido referido na cláusula FOREIGN_KEY seja explicitamente definido como em minúsculas ou composto por letras maiúsculas e minúsculas na instrução CREATE NICKNAME, é necessário especificar o apelido em maiúsculas ao referir-se a este apelido na cláusula FOREIGN_KEY. Quando esta opção estiver configurada em uma coluna, nenhuma outra opção pode ser configurada na coluna.
INPUT_MODE	Especifica o modo de entrada para a coluna. Os valores válidos são CONFIG e FILE_INPUT. CONFIG trata o valor como o modo de entrada para uma coluna. FILE_INPUT especifica um arquivo que armazena o valor. O wrapper passa o valor especificado para o daemon de script.
POSITION	Especifica um valor inteiro para os parâmetros posicionais. Se o valor posicional for configurado como um número inteiro, então, essa entrada deve estar nessa posição na linha de comandos. Se esta opção for configurada, o comutador é inserido no local apropriado quando a consulta é executada. Se POSITION for configurado como -1, a opção é incluída na última opção da linha de comandos. Um número inteiro POSITION não pode ser utilizado duas vezes no mesmo apelido. Esta opção aplica-se apenas às colunas de entrada.
PRIMARY_KEY	Obrigatório para um apelido pai que possui um ou mais apelidos filhos. Especifica que esse apelido é um apelido pai. O tipo de dados da coluna deve ser VARCHAR(16). Um apelido pode ter apenas uma opção de coluna PRIMARY_KEY. Yes é o único valor válido. Não especifique a opção XPATH para esta coluna. Esta coluna pode ser utilizada apenas para unir os apelidos pais aos apelidos filhos. Quando esta opção estiver configurada em uma coluna, nenhuma outra opção pode ser configurada na coluna.

Tabela 82. Opções de Colunas para Scripts (continuação)

Nome	Descrição
SWITCH	Especifica um sinalizador para o script na linha de comandos. O valor desta opção é precedido pelo valor da coluna, fornecido por WSSCRIPT.ARGS, ou pelo valor-padrão, se houver. Se você não especificar um valor para esta opção e um valor-padrão existir para a coluna, o valor-padrão é incluído sem informações sobre o comutador. Esta opção é obrigatória para colunas de entrada.
SWITCH_ONLY	Possibilita o uso de comutadores sem um argumento de linha de comandos. Os valores válidos são Y e N. Configure esta opção como Y e os valores de entrada válidos são Y e N. Para um valor de entrada Y, apenas o comutador é incluído na linha de comandos. Para um valor de entrada N, nenhum valor é incluído na linha de comandos.
VALID_VALUES	Especifica um conjunto de valores válidos para uma coluna. Utilize ponto e vírgula para separar vários valores.
XPATH	Especifica a expressão XPath no documento XML que contém os dados que correspondem a esta coluna. O wrapper avalia esta expressão XPath após a instrução CREATE NICKNAME aplicar a expressão XPath da opção de apelido XPATH.

Referência de Opções Sybase

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, o servidor, o mapeamento de usuário e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 83. Opções de Wrapper para Sybase

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. No Microsoft Windows, os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável. No UNIX, o valor padrão e o único valor válido é Y; o wrapper deve ser executado no modo em fence.

Tabela 83. Opções de Wrapper para Sybase (continuação)

Nome	Descrição
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Opções do Servidor

Tabela 84. Opções de Servidor para Sybase

Nome	Descrição
COLLATING_SEQUENCE	Especifica se a origem de dados especifica a mesma seqüência de intercalação padrão que o banco de dados federado. Os valores válidos são Y, N e I. I especifica a não distinção entre maiúsculas e minúsculas. O padrão é Y. A seqüência de intercalação especificada para o o servidor federado deve corresponder à seqüência de intercalação na origem de dados remota.
COMM_RATE	Especifica a taxa de comunicação, em megabytes por segundo, entre o servidor federado e o servidor de origem de dados. Os valores válidos são números inteiros maiores que 0 e menores que 2147483648. O padrão é 2.
CPU_RATIO	Especifica o quão mais rápida ou mais lenta a CPU de origem de dados é em comparação à CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de CPU; uma proporção 1:1. Uma configuração 0.5 indica que a CPU do servidor federado é 50% mais lenta que a CPU da origem de dados. Uma configuração 2 indica que a CPU federada é duas vezes mais rápida que a CPU da origem de dados.

Tabela 84. Opções de Servidor para Sybase (continuação)

Nome	Descrição
CONV_EMPTY_STRING	Especifica se o servidor federado converte uma cadeia vazia em um espaço durante as tarefas de replicação. Os valores válidos são Y e N. O padrão é N; o servidor federado não converte cadeias vazias. Configure esta opção como Y quando a origem de dados possui uma coluna de caractere não-anulável que armazena uma cadeia vazia.
DBNAME	Necessário. Especifica o nome do banco de dados que você deseja acessar. Obtenha o nome do banco de dados no servidor Sybase.
DB2_ID_QUOTE_CHAR	Specifies the quote characters that are used for delimited identifiers such as column names. If you do not specify this option, double quotes are used by default.
DB2_MAXIMAL_PUSHDOWN	Especifica os critérios primários que o otimizador de consultas utiliza para escolher um plano de acesso. Os valores válidos são Y e N. O padrão é N; o otimizador de consultas escolhe o plano que possui o menor custo estimado. Y especifica que o otimizador de consultas escolhe o plano de acesso que direciona a maioria das operações de consulta à origem de dados.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
DB2_TWO_PHASE_COMMIT	Especifica se o servidor federado se conecta à origem de dados no protocolo two-phase commit ou no protocolo one-phase commit. Os valores válidos são Y e N. O padrão é N; o servidor federado utiliza o protocolo one-phase commit para se conectar. Y especifica que o servidor federado utiliza o protocolo two-phase commit para se conectar.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.

Tabela 84. Opções de Servidor para Sybase (continuação)

Nome	Descrição
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.
FOLD_ID	Especifica o caso para o ID do usuário que é enviado à origem de dados. Não há valor-padrão; o servidor federado envia o ID do usuário em maiúsculas; em seguida, se o ID do usuário em maiúsculas falhar, o servidor envia o ID do usuário em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.
FOLD_PW	Especifica o caso para a senha que é enviada à origem de dados. Não há valor-padrão; o servidor federado envia a senha em maiúsculas; em seguida, se a senha em maiúsculas falhar, o servidor envia a senha em minúsculas. Os valores válidos são U (maiúscula), L (minúscula), e N (nulo). Evite utilizar a configuração nula, que pode resultar em desempenho insuficiente.
IFILE	Especifica o caminho e o nome do arquivo de interface do Sybase a ser utilizado em vez do arquivo de interface padrão. O wrapper Sybase procura pelo arquivo de interface nos seguintes locais, na ordem especificada: No Microsoft Windows, na opção do servidor IFILE, em seguida, no diretório %DB2PATH%\interfaces e, finalmente, no diretório %SYBASE%\ini\sql.ini. No UNIX, na opção do servidor IFILE, então no diretório sqllib/interfaces e, finalmente, no diretório \$SYBASE/interfaces.
IO_RATIO	Especifica o quão mais rápido ou mais lento o sistema I/O de origem de dados é executado em comparação ao sistema I/O do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de I/O; uma proporção 1:1. Uma configuração 0.5 indica que a velocidade do servidor federado é 50% mais lenta que a velocidade da origem de dados. Uma configuração 2 indica que a velocidade federada é duas vezes mais rápida que a velocidade da origem de dados.

Tabela 84. Opções de Servidor para Sybase (continuação)

Nome	Descrição
LOGIN_TIMEOUT	Especifica o tempo, em segundos, que o servidor federado aguarda antes de abandonar o pedido de login. O padrão é 0; o federated aguarda um período de tempo ilimitado.
NODE	Necessário. Especifica o nome do nó onde o servidor Sybase reside. O nome do nó está no arquivo de interfaces do Sybase.
OLD_NAME_GEN	Especifica como converter os nomes das colunas e os nomes de índices que estão na origem de dados nos nomes das colunas de apelidos e nomes de índices locais para o servidor federado. Os valores válidos são Y e N. O padrão é N; os nomes gerados correspondem fortemente aos nomes na origem de dados. Y especifica que os nomes gerados são os mesmos nomes criados no IBM WebSphere Federation Server Versão 9 e anterior. Assim, os nomes podem não corresponder tão fortemente aos nomes da origem de dados.
PACKET_SIZE	Especifica o tamanho do pacote, em bytes, que a biblioteca cliente utiliza para enviar pacotes TDS (Tabular Data Stream). Se o wrapper do Sybase precisa enviar ou receber grandes quantidades de dados de texto ou de imagem, aumente PACKET_SIZE.
PLAN_HINTS	Especifica se as sugestões de planos devem ou não ser ativadas. As sugestões de planos são fragmentos de instruções que fornecem informações adicionais que o otimizador de origem de dados utiliza para melhorar o desempenho das consultas. O otimizador de origem de dados utiliza as sugestões de planos para decidir se deve ou não utilizar um índice e qual índice ou qual sequência de junção de tabelas deve utilizar. Os valores válidos são Y e N. O padrão é N; as sugestões de planos não devem ser ativadas. Y especifica que as sugestões de planos devem ser ativadas na origem de dados, se suportar sugestões de planos.
PUSHDOWN	Especifica se o servidor federado permite que a origem de dados avalie as operações. Os valores válidos são Y e N. O padrão é Y; a origem de dados avalia as operações. N especifica que o servidor federado envia instruções SQL que incluem apenas SELECT com os nomes das colunas. Predicados, como a coluna WHERE=;, as funções escalares, como MAX e MIN;, as classificações, como ORDER BY OR GROUP BY;, e as junções não são incluídas no SQL que o servidor federado envia à origem de dados.

Tabela 84. Opções de Servidor para Sybase (continuação)

Nome	Descrição
SERVER_PRINCIPAL_NAME	Specifies the principal name of the Sybase server when the principal name is different from node name. The default value is the node name.
TIMEOUT	Especifica o tempo máximo, em segundos, que o servidor federado aguarda por uma resposta do servidor remoto a um comando. O padrão é 0, que especifica um período de tempo ilimitado.
XA_OPEN_STRING_OPTIONS	Especifica cadeias abertas para a interface Sybase DTM XA. Essas cadeias são adicionais ao nome LRM, nome de usuário e senha.

Opções de Mapeamento de Usuários

Tabela 85. Opções de Mapeamento de Usuários para Sybase

Opção	Descrição
ENABLE_KERBEROS_CONNECTION	Specifies the Kerberos security mechanism. The default value N indicates use of a default security mechanism. The value Y indicates that the Kerberos security mechanism is enabled.
REMOTE_AUTHID	Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.

Opções de Coluna

Tabela 86. Opções de Colunas para Sybase

Opção	Descrição
NUMERIC_STRING	Especifica como tratar as cadeias numéricas. O padrão é N. Se a coluna de cadeias da origem de dados contiver apenas cadeias numéricas e nenhum outro caractere, incluindo caracteres em branco, configure a opção NUMERIC_STRING para Y. Quando NUMERIC_STRING é configurado como Y para uma coluna, o otimizador de consultas reconhece que a coluna não contém caracteres em branco que poderiam interferir com a classificação dos dados na coluna. Utilize esta opção quando a sequência de intercalação de uma origem de dados for diferente da sequência de intercalação que o servidor federado utiliza. As colunas que utilizam esta opção não são excluídas da avaliação remota, devido a uma sequência de intercalação diferente.

Referência de Opções Teradata

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, o servidor, o mapeamento de usuário e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 87. Opções de Wrapper para Teradata

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. No Microsoft Windows, os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável. No UNIX, o valor-padrão é Y; o wrapper deve ser executado no modo em fence.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Opções do Servidor

Tabela 88. Opções de Servidor para Teradata

Nome	Descrição
COLLATING_SEQUENCE	Especifica se a origem de dados especifica a mesma seqüência de intercalação padrão que o banco de dados federado. Os valores válidos são Y, N e I. I especifica a não distinção entre maiúsculas e minúsculas. O padrão é Y. A seqüência de intercalação especificada para o o servidor federado deve corresponder à seqüência de intercalação na origem de dados remota.
COMM_RATE	Especifica a taxa de comunicação, em megabytes por segundo, entre o servidor federado e o servidor de origem de dados. Os valores válidos são números inteiros maiores que 0 e menores que 2147483648. O padrão é 2.
CPU_RATIO	Especifica o quão mais rápida ou mais lenta a CPU de origem de dados é em comparação à CPU do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de CPU; uma proporção 1:1. Uma configuração 0.5 indica que a CPU do servidor federado é 50% mais lenta que a CPU da origem de dados. Uma configuração 2 indica que a CPU federada é duas vezes mais rápida que a CPU da origem de dados.
DB2_MAXIMAL_PUSHDOWN	Especifica os critérios primários que o otimizador de consultas utiliza para escolher um plano de acesso. Os valores válidos são Y e N. O padrão é N; o otimizador de consultas escolhe o plano que possui o menor custo estimado. Y especifica que o otimizador de consultas escolhe o plano de acesso que direciona a maioria das operações de consulta à origem de dados.
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.

Tabela 88. Opções de Servidor para Teradata (continuação)

Nome	Descrição
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.
IO_RATIO	Especifica o quão mais rápido ou mais lento o sistema I/O de origem de dados é executado em comparação ao sistema I/O do servidor federado. Os valores válidos são maiores que 0 e menores que 1×10^{23} . O padrão é 1.0. Os valores podem ser expressados em qualquer notação dupla válida, por exemplo 123E10, 123, ou 1.21E4. Uma configuração 1 indica que o servidor federado e o servidor de origem de dados possuem a mesma velocidade de I/O; uma proporção 1:1. Uma configuração 0.5 indica que a velocidade do servidor federado é 50% mais lenta que a velocidade da origem de dados. Uma configuração 2 indica que a velocidade federada é duas vezes mais rápida que a velocidade da origem de dados.
NODE	Necessário. Especifica o nome de alias ou endereço IP do servidor Teradata.
OLD_NAME_GEN	Especifica como converter os nomes das colunas e os nomes de índices que estão na origem de dados nos nomes das colunas de apelidos e nomes de índices locais para o servidor federado. Os valores válidos são Y e N. O padrão é N; os nomes gerados correspondem fortemente aos nomes na origem de dados. Y especifica que os nomes gerados são os mesmos nomes criados no IBM WebSphere Federation Server Versão 9 e anterior. Assim, os nomes podem não corresponder tão fortemente aos nomes da origem de dados.

Tabela 88. Opções de Servidor para Teradata (continuação)

Nome	Descrição
PUSHDOWN	Especifica se o servidor federado permite que a origem de dados avalie as operações. Os valores válidos são Y e N. O padrão é Y; a origem de dados avalia as operações. N especifica que o servidor federado envia instruções SQL que incluem apenas SELECT com os nomes das colunas. Predicados, como a coluna WHERE=;, as funções escalares, como MAX e MIN;,, as classificações, como ORDER BY OR GROUP BY;,, e as junções não são incluídas no SQL que o servidor federado envia à origem de dados.

Opções de Mapeamento de Usuários

Tabela 89. Opções de Mapeamento de Usuários para Teradata

Nome	Descrição
REMOTE_AUTHID	Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.

Opções de Coluna

Tabela 90. Opções de Colunas para Teradata

Nome	Descrição
NUMERIC_STRING	Especifica como tratar as cadeias numéricas. O padrão é N. Se a coluna de cadeias da origem de dados contiver apenas cadeias numéricas e nenhum outro caractere, incluindo caracteres em branco, configure a opção NUMERIC_STRING para Y. Quando NUMERIC_STRING é configurado como Y para uma coluna, o otimizador de consultas reconhece que a coluna não contém caracteres em branco que poderiam interferir com a classificação dos dados na coluna. Utilize esta opção quando a sequência de intercalação de uma origem de dados for diferente da sequência de intercalação que o servidor federado utiliza. As colunas que utilizam esta opção não são excluídas da avaliação remota, devido a uma sequência de intercalação diferente.

Referência das Opções de Arquivos Estruturados em Tabelas

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique as opções de wrapper, servidor, apelido e de coluna.

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Opções de Wrapper

Tabela 91. Opções de Wrapper para Arquivos Estruturados em Tabela

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.

Opções do Servidor

Tabela 92. Opções de Servidor para Arquivos Estruturados em Tabela

Nome	Descrição
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.

Opções de Apelido

Tabela 93. Opções de Apelido para Arquivos Estruturados em Tabela

Nome	Descrição
COLUMN_DELIMITER	Especifica um caractere único para utilizar como delimitador que separa colunas no arquivo estruturado em tabela. O padrão é uma vírgula (,). Aspas simples não podem ser utilizadas como um delimitador. O delimitador de coluna deve ser consistente em todo o arquivo. Um valor nulo é representado por dois delimitadores próximos entre si ou por um delimitador seguido por um terminador de linha, se o campo NULL for o último na linha. O delimitador da coluna não pode existir como dados válidos para um coluna.
CODEPAGE	Especifica a página de códigos do arquivo na origem de dados. Esta opção é válida apenas para os bancos de dados federados que utilizam Unicode. Os dados de origem são convertidos da página de códigos especificada para Unicode.

Tabela 93. Opções de Apelido para Arquivos Estruturados em Tabela (continuação)

Nome	Descrição
FILE_PATH	Especifica o caminho completo para o arquivo estruturado em tabela. Feche o nome do arquivo em aspas simples. O arquivo de dados deve ser um arquivo padrão ou um link simbólico, em vez de um canal ou outro tipo de arquivo não padrão. Importante: Se você especificar a opção FILE_PATH, não especifique uma coluna DOCUMENT.
KEY_COLUMN	especifica o nome da coluna na qual o arquivo é classificado. Uma coluna que possui a opção de coluna DOCUMENT não pode ser a coluna-chave. Apenas as chaves de coluna únicas são suportadas. O valor deve ser o nome de uma coluna, definido na instrução CREATE NICKNAME. A coluna deve ser classificada em ordem crescente. A coluna-chave deve ser designada como não-anulável, incluindo a opção NOT NULL para sua definição na instrução de apelido. Este valor faz distinção entre maiúsculas e minúsculas.
SORTED	Especifica se o arquivo na origem de dados é ou não classificado na ordem crescente. Os valores válidos são Y e N. O padrão é N; o arquivo na origem de dados não é classificado em ordem crescente. As origens de dados classificadas devem ser classificadas em ordem crescente de acordo com a sequência de ordenação para o código do idioma atual, como definido pelas configurações na categoria LC_COLLATE National Language Support. Se você especificar que a origem de dados está classificada, configure a opção VALIDATE_DATA_FILE como Y.
VALIDATE_DATA_FILE	Para os arquivos classificados, especifica se o wrapper verifica que a coluna-chave é classificada em ordem crescente e verifica as chaves nulas. Esta validação ocorre apenas uma vez, quando o apelido é criado. O padrão é N; a ordem de classificação não é verificada. Esta opção é válida apenas quando a opção SORTED é configurada Y e a opção DOCUMENT não é especificada.

Opções de Coluna

Tabela 94. Opções de Colunas para Arquivos Estruturados em Tabela

Opção	Descrição
DOCUMENT	Permite especificar o caminho de arquivo quando a consulta é executada, em vez de ao criar o apelido. O único valor válido é FILE. Apenas uma coluna de cada apelido pode ser especificada com a opção DOCUMENT. A coluna associada à opção DOCUMENT deve ser um tipo de dado VARCHAR ou CHAR. Utilizar a opção de coluna de apelido DOCUMENT em vez da opção de apelido FILE_PATH implica que o arquivo que corresponde a este apelido será fornecido ao executar a consulta. Se a opção DOCUMENT possuir o valor FILE, o valor fornecido ao executar a consulta é o caminho completo do arquivo, cujo esquema corresponde à definição de apelido para este apelido.

Referência de Opções de Serviços da Web

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, servidor, mapeamento de usuário, apelido e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 95. Opções de Wrapper para Serviços da Web

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. O padrão é N; o wrapper é executado no modo confiável.
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.

Tabela 95. Opções de Wrapper para Serviços da Web (continuação)

Nome	Descrição
PROXY_TYPE	Especifica o tipo de proxy a ser utilizado para acessar a Internet, quando o servidor federado está protegido por firewall. Os valores válidos são NONE, HTTP e SOCKS. O valor-padrão é NONE.
PROXY_SERVER_NAME	Especifica o nome ou endereço IP do servidor proxy. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.
PROXY_SERVER_PORT	Especifica a porta ou o nome do serviço para o serviço de proxy no servidor proxy. Os valores válidos são um número da porta decimal de 1 a 32760 ou um nome do serviço.
SSL_KEYSTORE_FILE	Especifica o arquivo de armazenamento de certificados para as comunicações que utilizam o SSL ou TSL. Um valor válido é o nome do caminho completo acessível ao agente de banco de dados federado ou ao processo de modo em fence. O padrão é <i>install path/cfg/WSWrapperKeystore.kdb</i> .
SSL_KEYSTORE_PASSWORD	Especifica a senha a ser utilizada para acessar o arquivo na opção SSL_KEYSTORE_FILE. Os valores válidos são uma senha criptografada, ao ser armazenada no catálogo do banco de dados federado e <i>file:file_name</i> , em que <i>file_name</i> é o caminho completo para um arquivo stash.
SSL_VERIFY_SERVER_CERTIFICATE	Especifica o certificado do servidor é verificado durante a autenticação SSL. Os valores válidos são Y e N. O padrão é N; o certificado não é verificado.

Opções do Servidor

Tabela 96. Opções de Servidor para Serviços da Web

Nome	Descrição
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.

Tabela 96. Opções de Servidor para Serviços da Web (continuação)

Nome	Descrição
DB2_UM_PLUGIN	Especifica a implementação do plug-in de mapeamento de usuário. Para um plug-in gravado em Java, especifica uma cadeia com distinção entre maiúsculas e minúsculas para o nome de classe que corresponde à classe de repositório de mapeamento de usuário. Por exemplo, "UserMappingRepositoryLDAP". Para um plug-in gravado em C, especifica qualquer nome da biblioteca C válido.
DB2_UM_PLUGIN_LANG	Especifica o idioma do plug-in de mapeamento de usuário. Os valores válidos são Java e C. O padrão é Java.
PROXY_AUTHID	Especifica o nome de usuário para a autenticação do servidor proxy.
PROXY_PASSWORD	Especifica a senha para a autenticação do servidor proxy.
PROXY_SERVER_NAME	Especifica o nome ou endereço IP do servidor proxy. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.
PROXY_SERVER_PORT	Especifica a porta ou o nome do serviço para o serviço de proxy no servidor proxy. Os valores válidos são um número da porta decimal de 1 a 32760 ou um nome do serviço.
PROXY_TYPE	Especifica o tipo de proxy a ser utilizado para acessar a Internet, quando o servidor federado está protegido por firewall. Os valores válidos são NONE, HTTP e SOCKS. O valor-padrão é NONE.
SSL_CLIENT_CERTIFICATE_LABEL	Especifica o certificado de cliente a ser enviado durante a autenticação SSL. Se um valor não for especificado, o ID da autorização atual para o banco de dados federado é utilizado para localizar o certificado.
SSL_KEYSTORE_FILE	Especifica o arquivo de armazenamento de certificados para as comunicações que utilizam o SSL ou TSL. Um valor válido é o nome do caminho completo acessível ao agente de banco de dados federado ou ao processo de modo em fence. O padrão é <i>install path/cfg/WSWrapperKeystore.kdb</i> .

Tabela 96. Opções de Servidor para Serviços da Web (continuação)

Nome	Descrição
SSL_KEYSTORE_PASSWORD	Especifica a senha a ser utilizada para acessar o arquivo na opção SSL_KEYSTORE_FILE. Os valores válidos são uma senha criptografada, ao ser armazenada no catálogo do banco de dados federado e <code>file:file_name</code> , em que <code>file_name</code> é o caminho completo para um arquivo stash.
SSL_VERIFY_SERVER_CERTIFICATE	Especifica se deve verificar o certificado do servidor durante a autenticação SSL. Os valores válidos são Y e N. O padrão é N; o certificado não é verificado.

Opções de Mapeamento de Usuários

Tabela 97. Opções de Mapeamento de Usuários para Serviços da Web

Nome	Descrição
PROXY_AUTHID	Especifica o nome de usuário para a autenticação do servidor proxy.
PROXY_PASSWORD	Especifica a senha para a autenticação do servidor proxy. A senha é criptografada quando é armazenada no catálogo de banco de dados federado.
REMOTE_AUTHID	Especifica o ID do usuário remoto ao qual o ID do usuário local é mapeado. Se esta opção não for especificada, o ID utilizado para conectar-se ao banco de dados federado é utilizado.
REMOTE_PASSWORD	Especifica a senha remota para o ID do usuário remoto. Se esta opção não for especificada, a senha utilizada para conectar-se ao banco de dados federado é utilizada.
SSL_CLIENT_CERTIFICATE_LABEL	Especifica o certificado de cliente a ser enviado durante a autenticação SSL. Se um valor não for especificado, o ID de autorização atual do banco de dados federado é utilizado para localizar o certificado.

Opções de Apelido

Tabela 98. Opções de Apelido para Serviços da Web

Nome	Descrição
NAMESPACES	<p>Especifica os espaços de nomes associados aos prefixos de espaço de nomes utilizados nas opções XPATH e TEMPLATE para cada coluna. Utilize esta sintaxe:</p> <pre>NAMESPACES'prefix1="actual_namespace1", prefix2="actual_namespace2"'</pre> <p>Utilize uma vírgula para separar vários espaços de nomes. Por Exemplo:</p> <pre>NAMESPACES='http://www.myweb.com/cust", i='http://www.myweb.com/cust/id", n='http://www.myweb.com/cust/name"'</pre>
SOAPACTION	<p>Necessário para o apelido raiz. Especifica o atributo URI SOAPACTION do formato WSDL (Web Services Description Language). O URL pode conter um endereço IPv6 separado por vírgulas, se estiver encerrado por colchetes. Por exemplo: <code>http://[1080:0:0:0:8:800:200C:417A]</code></p> <p>Nota: Esta opção não é válida para apelidos não-raiz.</p>
STREAMING	<p>Especifica se o documento de origem deve ser separado em fragmentos lógicos para processamento. Os fragmentos correspondem ao nó que corresponde à expressão XPath do apelido. O wrapper então analisa e processa a origem de dados, fragmento por fragmento. Este tipo de análise minimiza o uso da memória. Os valores válidos são Y e N. O padrão é N; os documentos não são analisados. Esta opção é válida apenas em um apelido raiz.</p>
TEMPLATE	<p>Especifica o fragmento do modelo de apelido a ser utilizado para construir um pedido SOAP. O fragmento deve estar em conformidade com a sintaxe modelo especificada. Esta opção é válida apenas para o apelido raiz.</p>
URL	<p>Necessário para o apelido raiz. Especifica o URL do terminal em serviço da Web. Os protocolos suportados são HTTP e HTTPS. O URL pode conter um endereço IPv6 separado por vírgulas, se estiver encerrado por colchetes. Por exemplo: <code>http://[1080:0:0:0:8:800:200C:417A]</code></p>
XML_CODESET	<p>Especifica a codificação a ser utilizada para enviar e receber dados XML. Esta opção substitui a codificação interna.</p>

Tabela 98. Opções de Apelido para Serviços da Web (continuação)

Nome	Descrição
XPATH	Necessário. Especifica a expressão Xpath que identifica os elementos de resposta SOAP que representam tuples individuais. A expressão Xpath é utilizada como um contexto para a avaliação dos valores de colunas que são identificados pelas opções de coluna de apelido XPath.

Opções de Coluna

Tabela 99. Opções de Coluna para Serviços da Web

Nome	Descrição
ESCAPE_INPUT	Especifica se os caracteres especiais XML são substituídos nos valores de entrada XML ou não. Utilize esta opção para incluir fragmentos do XML como entrada, por exemplo, para incluir fragmentos do XML que possuem elementos de repetição. Os valores válidos são Y e N. N é o padrão; os valores de entrada são retidos. O tipo de dados da coluna deve ser VARCHAR ou CHAR. Se ESCAPE_INPUT for configurado como Y, é necessário também especificar a opção de coluna TEMPLATE.
FOREIGN_KEY	Indica que este apelido é um apelido filho e especifica o nome do apelido pai correspondente. Um apelido pode ter no máximo uma opção de coluna FOREIGN_KEY. O valor para a opção faz distinção entre maiúsculas e minúsculas. Não especifique a opção XPATH para esta coluna. A coluna pode ser utilizada apenas para unir um apelido pai e um apelido filho. Uma instrução CREATE NICKNAME que inclui uma opção FOREIGN_KEY falha, se o apelido pai possuir um nome do esquema diferente. A menos que o apelido referido na cláusula FOREIGN_KEY seja explicitamente definido como em minúsculas ou composto por letras maiúsculas e minúsculas na instrução CREATE NICKNAME, é necessário especificar o apelido em maiúsculas ao referir-se a este apelido na cláusula FOREIGN_KEY. Nota: <ul style="list-style-type: none"> • Quando esta opção estiver configurada em uma coluna, nenhuma outra opção pode ser configurada na coluna. • Se você configurar esta opção de coluna, não é possível utilizar posteriormente a instrução ALTER NICKNAME para descartar a opção. Em vez disso, é necessário descartar o apelido e, então, criar o apelido novamente sem essa opção de coluna.

Tabela 99. Opções de Coluna para Serviços da Web (continuação)

Nome	Descrição
PRIMARY_KEY	<p>Obrigatório para um apelido pai que possui um ou mais apelidos filhos. Especifica que esse apelido é um apelido pai. O tipo de dados da coluna deve ser VARCHAR(16). Um apelido pode ter apenas uma opção de coluna PRIMARY_KEY. Yes é o único valor válido. Não especifique a opção XPATH para esta coluna. Esta coluna pode ser utilizada apenas para unir os apelidos pais aos apelidos filhos.</p> <p>Nota:</p> <ul style="list-style-type: none"> • Quando esta opção estiver configurada em uma coluna, nenhuma outra opção pode ser configurada na coluna. • Se você configurar esta opção de coluna, não é possível utilizar posteriormente a instrução ALTER NICKNAME para descartar a opção. Em vez disso, é necessário descartar o apelido e, então, criar o apelido novamente sem essa opção de coluna.
SOAPACTIONCOLUMN	<p>Especifica a coluna que especifica dinamicamente a ação do SOAP para o terminal de serviços da Web ao executar uma consulta. Esta opção é válida apenas para o apelido raiz. Se o nome do host for um endereço IPv6 (separado por vírgulas), feche o nome do host em colchetes. Por exemplo: 'http://[1080:0:0:0:8:800:200C:417A]:99/soap'</p> <p>Quando esta opção está configurada em uma coluna, nenhuma outra opção pode ser configurada na coluna.</p>
TEMPLATE	<p>Especifica o fragmento do modelo de coluna a ser utilizado para construir o documento de entrada XML. O fragmento deve estar em conformidade com a sintaxe modelo especificada.</p> <p>Nota: Se você configurar esta opção de coluna, não é possível utilizar posteriormente a instrução ALTER NICKNAME para descartar a opção. Em vez disso, é necessário descartar o apelido e, então, criar o apelido novamente sem essa opção de coluna.</p>

Tabela 99. Opções de Coluna para Serviços da Web (continuação)

Nome	Descrição
URLCOLUMN	Especifica a coluna que especifica dinamicamente a ação do SOAP para o terminal de serviços da Web ao executar uma consulta. Esta opção é válida apenas para o apelido raiz. Se o nome do host for um endereço IPv6 (separado por vírgulas), feche o nome do host em colchetes. Por exemplo: 'http://[1080:0:0:0:8:800:200C:417A]:99/soap' Quando esta opção está configurada em uma coluna, nenhuma outra opção pode ser configurada na coluna.
XPATH	Especifica a expressão XPath no documento XML que contém os dados que correspondem a esta coluna. O wrapper avalia esta expressão XPath após a instrução CREATE NICKNAME aplicar a expressão XPath da opção de apelido XPATH. Nota: Se você configurar esta opção de coluna, não é possível utilizar posteriormente a instrução ALTER NICKNAME para descartar a opção. Em vez disso, é necessário descartar o apelido e, então, criar o apelido novamente sem essa opção de coluna.

Referência de Opções XML

Para configurar como o servidor federado e seus usuários interagem com uma origem de dados, configure e modifique o wrapper, servidor, mapeamento de usuário, apelido e as opções de coluna.

Opções de Wrapper

As tabelas a seguir listam as opções que se aplicam a essa origem de dados e identificam as opções necessárias que devem ser especificadas.

Tabela 100. Opções de Wrapper para XML

Nome	Descrição
DB2_FENCED	Necessário. Especifica se o wrapper é executado no modo em fence ou no modo confiável. Os valores válidos são Y e N. N é o padrão; o wrapper é executado em modo confiável.
PROXY_TYPE	Especifica o tipo de proxy a ser utilizado para acessar a Internet, quando o servidor federado está protegido por firewall. Os valores válidos são NONE, HTTP e SOCKS. O valor-padrão é NONE.

Tabela 100. Opções de Wrapper para XML (continuação)

Nome	Descrição
PROXY_SERVER_NAME	Especifica o nome ou endereço IP do servidor proxy. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.
PROXY_SERVER_PORT	Especifica a porta ou o nome do serviço para o serviço de proxy no servidor proxy. Os valores válidos são um número da porta decimal de 1 a 32760 ou um nome do serviço.
SSL_KEYSTORE_FILE	Especifica o arquivo de armazenamento de certificados para as comunicações que utilizam o SSL ou TSL. Um valor válido é o nome do caminho completo acessível ao agente de banco de dados federado ou ao processo de modo em fence.
SSL_KEYSTORE_PASSWORD	Especifica a senha a ser utilizada para acessar o arquivo na opção SSL_KEYSTORE_FILE. Os valores válidos são uma senha criptografada, ao ser armazenada no catálogo do banco de dados federado e <code>file:file_name</code> , em que <code>file_name</code> é o caminho completo para um arquivo stash.
SSL_VERIFY_SERVER_CERTIFICATE	Especifica o certificado do servidor é verificado durante a autenticação SSL. Os valores válidos são Y e N. O padrão é N; o certificado não é verificado.

Opções do Servidor

Tabela 101. Opções de Servidor para XML

Nome	Descrição
DB2_MAX_ASYNC_REQUESTS_PER_QUERY	Especifica o número máximo de pedidos assíncronos simultâneos de uma consulta. Os valores válidos são de -1 a 64000. O padrão é 1. -1 especifica que o otimizador de consultas federadas determina o número de pedidos. 0 especifica que a origem de dados não pode acomodar pedidos assíncronos adicionais.
PROXY_AUTHID	Especifica o nome de usuário para a autenticação do servidor proxy.
PROXY_PASSWORD	Especifica a senha para a autenticação do servidor proxy.
PROXY_SERVER_NAME	Especifica o nome ou endereço IP do servidor proxy. Os endereços IP válidos estão no formato IPv4 (separados por pontos) ou no formato IPv6 (separados por vírgulas). Utilize o formato IPv6 apenas se IPv6 estiver configurado.

Tabela 101. Opções de Servidor para XML (continuação)

Nome	Descrição
PROXY_SERVER_PORT	Especifica a porta ou o nome do serviço para o serviço de proxy no servidor proxy. Os valores válidos são um número da porta decimal de 1 a 32760 ou um nome do serviço.
PROXY_TYPE	Especifica o tipo de proxy a ser utilizado para acessar a Internet, quando o servidor federado está protegido por firewall. Os valores válidos são NONE, HTTP e SOCKS. O valor-padrão é NONE.
SOCKET_TIMEOUT	Especifica o tempo máximo, em minutos, que o servidor federado aguarda pelos resultados do servidor proxy. Um valor válido é qualquer número que é maior ou igual a 0. O padrão é 0; o servidor aguarda um período ilimitado de tempo.
SSL_CLIENT_CERTIFICATE_LABEL	Especifica o certificado de cliente a ser enviado durante a autenticação SSL. Se um valor não for especificado, o ID da autorização atual para o banco de dados federado é utilizado para localizar o certificado.
SSL_KEYSTORE_FILE	Especifica o arquivo de armazenamento de certificados para as comunicações que utilizam o SSL ou TSL. Um valor válido é o nome do caminho completo acessível ao agente de banco de dados federado ou ao processo de modo em fence.
SSL_KEYSTORE_PASSWORD	Especifica a senha a ser utilizada para acessar o arquivo na opção SSL_KEYSTORE_FILE. Os valores válidos são uma senha criptografada, ao ser armazenada no catálogo do banco de dados federado e <code>file:file_name</code> , em que <code>file_name</code> é o caminho completo para um arquivo stash.
SSL_VERIFY_SERVER_CERTIFICATE	Especifica se deve verificar o certificado do servidor durante a autenticação SSL. O padrão é N; o certificado não é verificado.

Opções de Mapeamento de Usuários

Tabela 102. Opções de Mapeamento de Usuários para XML

Nome	Descrição
PROXY_AUTHID	Especifica o nome de usuário para a autenticação do servidor proxy.
PROXY_PASSWORD	Especifica a senha para a autenticação do servidor proxy.

Tabela 102. Opções de Mapeamento de Usuários para XML (continuação)

Nome	Descrição
SSL_CLIENT_CERTIFICATE_LABEL	Especifica o certificado de cliente a ser enviado durante a autenticação SSL. Se um valor não for especificado, o ID da autorização atual para o banco de dados federado é utilizado para localizar o certificado.

Opções de Apelido

Tabela 103. Opções de Apelidos para XML

Nome	Descrição
DIRECTORY_PATH	Especifica o nome do caminho de um diretório que contém um ou mais arquivos XML. Utilize esta opção para criar um apelido único em vários arquivos de origem XML. O wrapper XML utiliza apenas os arquivos com uma extensão .xml que estão localizados no diretório especificado. O wrapper XML ignora todos os outros arquivos neste diretório. Se você especificar esta opção de apelido, não especifique uma coluna DOCUMENT. Esta opção é válida apenas para o apelido raiz.
FILE_PATH	Especifica o caminho de arquivo do documento XML. Se você especificar FILE_PATH, não especifique uma coluna DOCUMENT. Esta opção é válida apenas para o apelido raiz.
INSTANCE_PARSE_TIME	Especifica o tempo, em milissegundos, que é necessário para analisar uma linha do documento de origem XML. O valor válido pode ser um valor inteiro ou decimal. O padrão é 7. Esta opção é válida apenas para colunas do apelido raiz. Para otimizar consultas de estruturas de origem XML grandes ou complexas, modifique as opções INSTANCE_PARSE_TIME, XPATH_EVAL_TIME e NEXT_TIME.
NAMESPACES	Especifica os espaços de nomes associados aos prefixos de espaço de nomes utilizados nas opções XPATH e TEMPLATE para cada coluna. Utilize esta sintaxe: <code>NAMESPACES 'prefix1="actual_namespace1", prefix2="actual_namespace2"'</code> Utilize uma vírgula para separar vários espaços de nomes. Por Exemplo: <code>NAMESPACES='http://www.myweb.com/cust', i='http://www.myweb.com/cust/id', n='http://www.myweb.com/cust/name''</code>

Tabela 103. Opções de Apelidos para XML (continuação)

Nome	Descrição
NEXT_TIME	Especifica o tempo, em milissegundos, que é necessário para localizar elementos de origem subsequentes da expressão XPath. O padrão é 1. Esta opção é válida para os apelidos raiz e apelidos não-raiz. Para otimizar consultas de estruturas de origem XML grandes ou complexas, modifique as opções INSTANCE_PARSE_TIME, XPATH_EVAL_TIME e NEXT_TIME.
STREAMING	Especifica se o documento de origem deve ser separado em fragmentos lógicos para processamento. Os fragmentos correspondem ao nó que corresponde à expressão XPath do apelido. O wrapper então analisa e processa a origem de dados, fragmento por fragmento. Este tipo de análise minimiza o uso da memória. Os valores válidos são Y e N. O padrão é N; os documentos não são analisados. Esta opção é válida apenas em um apelido raiz.
VALIDATE	Especifica se o documento de origem é validado para assegurar sua conformidade com um esquema XML ou DTD (Document Type Definition) antes de extrair seus dados. O padrão é N; a validação não ocorre. Antes de configurar o valor como Y, o arquivo de esquema ou o arquivo DTD está no local que o documento de origem especifica. Esta opção é válida apenas para o apelido raiz. Não configure ambas as opções STREAMING e VALIDATE como Y.
XPATH	Necessário. Especifica a expressão Xpath que identifica os elementos que representam tuples individuais. A expressão Xpath é utilizada como um contexto para a avaliação dos valores de colunas que são identificados pela opção de coluna XPath.
XPATH_EVAL_TIME	Especifica o tempo necessário, em milissegundos, para avaliar a expressão XPath do apelido e localizar o primeiro elemento. O valor pode ser um número inteiro ou valor decimal. O padrão é 1. Esta opção é válida para os apelidos raiz e apelidos não-raiz. Para otimizar consultas de estruturas de origem XML grandes ou complexas, modifique as opções INSTANCE_PARSE_TIME, XPATH_EVAL_TIME e NEXT_TIME.

Opções de Coluna

Tabela 104. Opções de Colunas para XML

Nome	Descrição
DOCUMENT	<p>Especifica que esta coluna é uma coluna DOCUMENT. O valor da coluna DOCUMENT indica o tipo de dados de origem XML fornecido ao apelido ao executar a consulta. Esta opção é válida apenas para colunas do apelido raiz (o apelido que identifica os elementos de nível superior do documento XML). Apenas uma coluna para cada apelido pode ser especificada com a opção DOCUMENT. Se você utilizar a opção de coluna de apelido DOCUMENT em vez da opção de apelido FILE_PATH ou DIRECTORY_PATH, o documento que corresponde a este apelido será fornecido ao executar a consulta. Os valores válidos são FILE, DIRECTORY, URI e COLUMN. O padrão é FILE. A coluna deve ter o tipo de dado VARCHAR.</p>
FOREIGN_KEY	<p>Indica que este apelido é um apelido filho e especifica o nome do apelido pai correspondente. Um apelido pode ter no máximo uma opção de coluna FOREIGN_KEY. O valor para a opção faz distinção entre maiúsculas e minúsculas. Não especifique a opção XPATH para esta coluna. A coluna pode ser utilizada apenas para unir um apelido pai e um apelido filho. Uma instrução CREATE NICKNAME que inclui uma opção FOREIGN_KEY falha, se o apelido pai possuir um nome do esquema diferente. A menos que o apelido referido na cláusula FOREIGN_KEY seja explicitamente definido como em minúsculas ou composto por letras maiúsculas e minúsculas na instrução CREATE NICKNAME, é necessário especificar o apelido em maiúsculas ao referir-se a este apelido na cláusula FOREIGN_KEY.</p> <p>Nota:</p> <ul style="list-style-type: none">• Quando esta opção estiver configurada em uma coluna, nenhuma outra opção pode ser configurada na coluna.• Se você configurar esta opção de coluna, não é possível utilizar posteriormente a instrução ALTER NICKNAME para descartar a opção. Em vez disso, é necessário descartar o apelido e, então, criar o apelido novamente sem essa opção de coluna.

Tabela 104. Opções de Colunas para XML (continuação)

Nome	Descrição
PRIMARY_KEY	<p>Obrigatório para um apelido pai que possui um ou mais apelidos filhos. Especifica que esse apelido é um apelido pai. O tipo de dados da coluna deve ser VARCHAR(16). Um apelido pode ter apenas uma opção de coluna PRIMARY_KEY. Yes é o único valor válido. Não especifique a opção XPATH para esta coluna. Esta coluna pode ser utilizada apenas para unir os apelidos pais aos apelidos filhos.</p> <p>Nota:</p> <ul style="list-style-type: none"> • Quando esta opção estiver configurada em uma coluna, nenhuma outra opção pode ser configurada na coluna. • Se você configurar esta opção de coluna, não é possível utilizar posteriormente a instrução ALTER NICKNAME para descartar a opção. Em vez disso, é necessário descartar o apelido e, então, criar o apelido novamente sem essa opção de coluna.
XPATH	<p>Especifica a expressão XPath no documento XML que contém os dados que correspondem a esta coluna. O wrapper avalia esta expressão XPath após a instrução CREATE NICKNAME aplicar a expressão XPath da opção de apelido XPATH.</p> <p>Nota: Se você configurar esta opção de coluna, não é possível utilizar posteriormente a instrução ALTER NICKNAME para descartar a opção. Em vez disso, é necessário descartar o apelido e, então, criar o apelido novamente sem essa opção de coluna.</p>

Capítulo 5. Visualizações na Tabela de Catálogos Globais Contendo Informações sobre Federação

A maioria das visualizações de catálogo em um banco de dados federado são as mesmas das visualizações de catálogo em qualquer outro banco de dados DB2 para Linux, UNIX e Windows.

Há várias visualizações únicas que contêm informações pertinentes a um sistema federado, como a visualização SYSCAT.WRAPPERS.

As visualizações SYSCAT são somente leitura. Não é possível emitir uma atualização ou inserir a operação em uma visualização no esquema SYSCAT. Utilizar as visualizações SYSSTAT é a forma recomendada de atualizar o catálogo do sistema. Altere os aplicativos que referenciam a visualização SYSCAT para referenciar a visualização SYSSTAT atualizável.

A tabela a seguir lista as visualizações SYSCAT que contêm informações federadas. Essas são visualizações somente leitura.

Tabela 105. Visualizações do Catálogo Utilizadas Geralmente com um Sistema Federado

Exibições de Catálogos	Descrição
SYSCAT.CHECKS	Contém as informações de restrição de verificação definidas.
SYSCAT.COLCHECKS	Contém colunas referenciadas por uma restrição de verificação.
SYSCAT.COLUMNS	Contém as informações da coluna sobre os objetos da origem de dados (tabelas e visualizações) para os quais você criou apelidos.
SYSCAT.COLOPTIONS	Contém informações sobre os valores das opções de coluna às quais você definiu um apelido.
SYSCAT.CONSTDEP	Contém a dependência de uma restrição informativa definida.
SYSCAT.DATATYPES	Contém as informações de tipo de dado sobre os tipos de dados incorporados e definidos pelo usuário do DB2.
SYSCAT.DBAUTH	Contém as autoridades de bancos de dados mantidas por usuários individuais e grupos.
SYSCAT.FUNCMAPOPTIONS	Contém as informações sobre os valores de opções definidas para um mapeamento de função.
SYSCAT.FUNCMAPPINGS	Contém os mapeamentos de funções entre o banco de dados federado e os objetos da origem de dados.
SYSCAT.INDEXCOLUSE	Contém as colunas que participam em um índice.
SYSCAT.INDEXES	Contém as especificações de índice para os objetos da origem de dados.

Tabela 105. Visualizações do Catálogo Utilizadas Geralmente com um Sistema Federado (continuação)

Exibições de Catálogos	Descrição
SYSCAT.INDEXOPTIONS	Contém as informações sobre as opções de índice.
SYSCAT.KEYCOLUSE	Contém as colunas que participam em uma chave definida por uma restrição de chave exclusiva, chave primária ou chave estrangeira.
SYSCAT.NICKNAMES	Contém informações sobre os apelidos criados.
SYSCAT.REFERENCES	Contém informações sobre as restrições de referências definidas.
SYSCAT.ROUTINES	Contém funções locais definidas pelo usuário do DB2 ou modelos de funções. Os modelos de funções são utilizados para mapear uma função de origem de dados.
SYSCAT.REVTYPEMAPPINGS	Esta visualização não é utilizada. Todos os mapeamentos de tipos de dados são registrados na visualização SYSCAT.TYPEMAPPINGS.
SYSCAT.ROUTINEOPTIONS	Contém informações sobre os valores das opções de rotina federadas.
SYSCAT.ROUTINEPARMOPTIONS	Contém informações sobre os valores das opções de parâmetros de rotina federadas.
SYSCAT.ROUTINEPARMS	Contém um parâmetro ou o resultado de uma rotina definida em SYSCAT.ROUTINES.
SYSCAT.ROUTINESFEDERATED	Contém informações sobre as rotinas federadas definidas.
SYSCAT.SERVERS	Contém definições do servidor que você cria para os servidores de origem de dados.
SYSCAT.TABCONST	Cada linha representa uma tabela e limitadores de apelidos do tipo CHECK, UNIQUE, PRIMARY KEY ou FOREIGN KEY.
SYSCAT.TABLES	Contém informações sobre cada tabela local do DB2, visualização federada e apelido criados.
SYSCAT.TYPEMAPPINGS	Contém mapeamentos de tipos de dados de encaminhamento e mapeamentos de tipos de dados reversos. O mapeamento é para tipos de dados locais do DB2 de tipos de dados da origem de dados. Esses mapeamentos são utilizados ao criar um apelido em um objeto de origem de dados.
SYSCAT.USEROPTIONS	Contém as informações de autorização do usuário, configuradas ao criar mapeamentos de usuários entre o banco de dados federado e os servidores de origem de dados.
SYSCAT.VIEWS	Contém informações sobre as visualizações federadas locais criadas.
SYSCAT.WRAPOPTIONS	Contém informações sobre os valores de opções configuradas para um wrapper.

Tabela 105. Visualizações do Catálogo Utilizadas Geralmente com um Sistema Federado (continuação)

Exibições de Catálogos	Descrição
SYSCAT.WRAPPERS	Contém o nome do wrapper e o arquivo de biblioteca para cada origem de dados à qual você cria um wrapper.

A tabela a seguir lista as visualizações SYSSTAT que contém informações federadas. Essas são visualizações de leitura/gravação, que contém estatísticas que podem ser atualizadas.

Tabela 106. Visualizações do Catálogo Global Atualizáveis e Federadas

Exibições de Catálogos	Descrição
SYSSTAT.COLUMNS	Contém as informações estatísticas sobre cada coluna nos objetos da origem de dados (tabelas e visualizações) para os quais você criou apelidos. As estatísticas não são gravadas para as colunas herdadas nas tabelas digitadas.
SYSSTAT.INDEXES	Contém informações estatísticas sobre cada especificação de índice para os objetos de origem de dados.
SYSSTAT.ROUTINES	Contém informações estatísticas sobre cada função definida pelo usuário. Não inclui funções integradas. As estatísticas não são gravadas para as colunas herdadas nas tabelas digitadas.
SYSSTAT.TABLES	Contém informações sobre cada tabela de base. As informações de visualização, sinônimos e alias não estão incluídas nesta visualização. Para tabelas digitadas, apenas a tabela raiz de uma hierarquia de tabelas é incluída na visualização. As estatísticas não são gravadas para as colunas herdadas nas tabelas digitadas.

Capítulo 6. Opções de Mapeamento de Funções para Sistemas Federados

O servidor federado fornece mapeamentos-padrão entre as funções do DB2 e as funções da origem de dados. Para a maioria das origens de dados, os mapeamentos de funções padrão estão nos wrappers. Para utilizar uma função de origem de dados que o servidor federado não reconhece ou alterar o mapeamento padrão, crie um mapeamento de função.

Ao criar um mapeamento de função, você especifica o nome da função da origem de dados e deve ativar a função mapeada. Em seguida, ao utilizar a função mapeada, o otimizador de consultas compara o custo da execução da função na origem de dados com o custo da execução da função no servidor federado.

Tabela 107. Opções para os Mapeamentos de Funções

Nome	Descrição
DISABLE	Ativa ou desativa um mapeamento de função padrão. Os valores válidos são Y e N. O padrão é N.
REMOTE_NAME	O nome da função de origem de dados. O padrão é o nome local.

Capítulo 7. Tipos de servidores válidos em instruções SQL

Os tipos de servidores indicam o tipo de origem de dados que a definição de servidor representa.

Os tipos de servidores variam por fornecedor, propósito e sistema operacional. Os valores suportados dependem da origem de dados.

Para a maioria das origens de dados, é necessário especificar um tipo de servidor válido na instrução CREATE SERVER.

Tabela 108. Origens de Dados e Tipos de Servidores

Origem de Dados	Tipo de Servidor
BioRS	Um tipo de servidor não é necessário na instrução CREATE SERVER.
Excel	Um tipo de servidor não é necessário na instrução CREATE SERVER.
IBM DB2 Universal Database para Linux, UNIX e Windows	DB2/UDB
IBM DB2 Universal Database para System i e AS/400	DB2/ISERIES
IBM DB2 Universal Database para z/OS	DB2/ZOS
IBM DB2 para VM	DB2/VM
Informix	INFORMIX
JDBC	JDBC (Obrigatório para origens de dados do JDBC que são suportadas pelos drivers JDBC 3.0 e posterior).
Microsoft SQL Server	MSSQLSERVER (Obrigatório para as origens de dados suportadas pelo driver DataDirect Connect ODBC 4.2 (ou posterior) ou o driver Microsoft SQL Server ODBC 3.0 (ou posterior).
ODBC	ODBC (Obrigatório para origens de dados do ODBC que são suportadas pelo driver ODBC 3.x).
OLE DB	Um tipo de servidor não é necessário na instrução CREATE SERVER.
Oracle	ORACLE (Obrigatório para origens de dados do Oracle suportadas pelo software cliente Oracle NET8).
Sybase (CTLIB)	SYBASE
Arquivos estruturados em tabelas	Um tipo de servidor não é necessário na instrução CREATE SERVER.
Teradata	TERADATA
Serviços da Web	Um tipo de servidor não é necessário na instrução CREATE SERVER.
XML	Um tipo de servidor não é necessário na instrução CREATE SERVER.

Capítulo 8. Mapeamentos de Tipos de Dados

Os mapeamentos de tipos de dados para as origens de dados relacionais incluem os mapeamentos de tipos de encaminhamento, mapeamentos de tipos reversos e mapeamentos de tipos que são específicos ao Unicode. Cada uma das origens de dados não relacionais suportam tipos de dados específicos.

Mapeamentos de tipo de dados de encaminhamento padrão

Os dois tipos de mapeamentos entre os tipos de dados de origem de dados e os tipos de dados do banco de dados federado são os mapeamentos de tipos de encaminhamento e os mapeamentos de tipos reversos. Em um mapeamento de tipo de encaminhamento, o mapeamento é de um tipo remoto a um tipo local comparável.

É possível substituir um mapeamento de tipo padrão ou criar um novo mapeamento de tipo com a instrução CREATE TYPE MAPPING.

Esses mapeamentos são válidos com todas as versões suportadas, a menos que seja observado de outra maneira.

Para todos os mapeamentos de tipos de dados de encaminhamento padrão de uma origem de dados do banco de dados federado, o esquema federado é SYSIBM.

As tabelas a seguir mostram os mapeamentos de encaminhamento padrão entre os tipos de dados de banco de dados federado e os tipos de dados de origem de dados.

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de dados do Banco de Dados DB2 para Linux, UNIX e Windows

A tabela a seguir lista o tipo de dados de encaminhamento padrão para origens de dados do Banco de dados DB2 para Linux, UNIX, e Windows.

Tabela 109. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para o Banco de Dados DB2 para Linux, UNIX, e Windows (Não são mostradas todas as colunas)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
BIGINT	-	-	-	-	-	-	BIGINT	-	0	-
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	-	-	-	-	-	-	CHAR	-	0	N
CHAR	-	-	-	-	S	-	CHAR	-	0	S
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DATE	-	-	-	-	-	-	TIMESTAMP ¹	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DECFLOAT ²	-	-	-	-	-	-	DECFLOAT	-	0	-
DOUBLE	-	-	-	-	-	-	DOUBLE	-	-	-

Tabela 109. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para o Banco de Dados DB2 paraLinux, UNIX, e Windows (Não são mostradas todas as colunas) (continuação)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
FLOAT	-	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
LONGVAR	-	-	-	-	N	-	CLOB	-	-	-
LONGVAR	-	-	-	-	S	-	BLOB	-	-	-
LONGVARG	-	-	-	-	-	-	DBCLOB	-	-	-
REAL	-	-	-	-	-	-	REAL	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP(p)	-	-	p	p	-	-	TIMESTAMP(p)	-	p	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	0	N
VARCHAR	-	-	-	-	S	-	VARCHAR	-	0	S
VARGRAPH	-	-	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	0	N

Nota:

1. O tipo federado é TIMESTAMP(0) se o parâmetro de configuração date_compat estiver definido como ON.
2. A opção do servidor SAME_DECFLT_ROUNDING é configurada como N por padrão e as operações não serão rebaixasadas à origem de dados remota, a menos que SAME_DECFLT_ROUNDING seja configurado como Y. Para obter informações sobre a opção do servidor SAME_DECFLT_ROUNDING, consulte as Referência das Opções do Banco de Dados DB2.

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do DB2 para System i

A tabela a seguir lista o tipo de dados de encaminhamento padrão para origens de dados do DB2 para System i.

Tabela 110. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para DB2 para System i (Não são mostradas todas as colunas)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	255	32672	-	-	-	-	VARCHAR	-	0	N
CHAR	1	254	-	-	S	-	CHAR	-	0	S
CHAR	255	32672	-	-	S	-	VARCHAR	-	0	S
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
GRAPHIC	128	16336	-	-	-	-	VARGRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-

Tabela 110. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para DB2 para System i (Não são mostradas todas as colunas) (continuação)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
NUMERIC	-	-	-	-	-	-	DECIMAL	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP(6)	-	6	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	S	-	VARCHAR	-	0	S
VARG	1	16336	-	-	-	-	VARGGRAPHIC	-	0	N
VARGGRAPHIC	1	16336	-	-	-	-	VARGGRAPHIC	-	0	N

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do DB2 para VM e VSE

A tabela a seguir lista o tipo de dados de encaminhamento padrão para origens de dados do DB2 para VM e VSE.

Tabela 111. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para DB2 Server para VM e VSE (Não são mostradas todas as colunas)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	1	254	-	-	S	-	CHAR	-	0	S
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBAHW	-	-	-	-	-	-	SMALLINT	-	0	-
DBAINT	-	-	-	-	-	-	INTEGER	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP(6)	-	6	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	S	-	VARCHAR	-	0	S
VARGGRAPHIC	1	16336	-	-	-	-	VARGGRAPHIC	-	0	N
VARGGRAPH	1	16336	-	-	-	-	VARGGRAPHIC	-	0	N

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do DB2 para z/OS

A tabela a seguir lista o tipo de dados de encaminhamento padrão para origens de dados do DB2 para z/OS.

Tabela 112. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para DB2 para z/OS (Não são mostradas todas as colunas)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
BIGINT	0	8	0	0	" "	"/0"	BIGINT	0	0	N
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	255	32672	-	-	-	-	VARCHAR	-	0	N
CHAR	1	254	-	-	S	-	CHAR	-	0	S
CHAR	255	32672	-	-	S	-	VARCHAR	-	0	S
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
ROWID	-	-	-	-	S	-	VARCHAR	40	-	S
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP(6)	-	6	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	S	-	VARCHAR	-	0	S
VARG	1	16336	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	0	N

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Informix

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do Informix.

Tabela 113. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Informix (Não são mostradas todas as colunas)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
BLOB	-	-	-	-	-	-	BLOB	2147483647	-	-
BOOLEAN	-	-	-	-	-	-	CHARACTER	1	-	-
BYTE	-	-	-	-	-	-	BLOB	2147483647	-	-
CHAR	1	254	-	-	-	-	CHARACTER	-	-	-
CHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
CLOB	-	-	-	-	-	-	CLOB	2147483647	-	-

Tabela 113. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Informix (Não são mostradas todas as colunas) (continuação)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
DATE	-	-	-	-	-	-	DATE	4	-	-
DATE	-	-	-	-	-	-	TIMESTAMP ¹	-	0	-
DATETIME ²	0	4	0	4	-	-	DATE	4	-	-
DATETIME	6	10	6	10	-	-	TIME	3	-	-
DATETIME	0	4	6	15	-	-	TIMESTAMP(6)	10	6	-
DATETIME	6	10	11	15	-	-	TIMESTAMP(6)	10	6	-
DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
DECIMAL	32	130	-	-	-	-	DOUBLE	8	-	-
DECIMAL	1	32	255	255	-	-	DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
INTERVAL	-	-	-	-	-	-	VARCHAR	25	-	-
INT8	-	-	-	-	-	-	BIGINT	19	0	-
LVARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
MONEY	1	31	0	31	-	-	DECIMAL	-	-	-
MONEY	32	32	-	-	-	-	DOUBLE	8	-	-
NCHAR	1	254	-	-	-	-	CHARACTER	-	-	-
NCHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
NVARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
REAL	-	-	-	-	-	-	REAL	4	-	-
SERIAL	-	-	-	-	-	-	INTEGER	4	-	-
SERIAL8	-	-	-	-	-	-	BIGINT	-	-	-
SMALLFLOAT	-	-	-	-	-	-	REAL	4	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
TEXT	-	-	-	-	-	-	CLOB	2147483647	-	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-

Notas:

1. O tipo federado é TIMESTAMP(0) se o parâmetro de configuração date_compat estiver definido como ON.
2. Para os tipos de dados Informix DATETIME, o servidor federado DB2 UNIX e Windows utiliza o qualificador de alto nível do Informix, como REMOTE_LENGTH, e o qualificador de baixo nível do Informix, como REMOTE_SCALE.

Os qualificadores do Informix são as constantes "TU_" definidas no arquivo Informix Client SDK datatime.h. As constantes são:

0 = YEAR	8 = MINUTE	13 = FRACTION(3)
2 = MONTH	10 = SECOND	14 = FRACTION(4)
4 = DAY	11 = FRACTION(1)	15 = FRACTION(5)
6 = HOUR	12 = FRACTION(2)	

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do JDBC

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do JDBC.

Tabela 114. Mapeamentos de Tipos de Dados de Encaminhamento Padrão JDBC

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
BIGINT	-	-	-	-	-	-	BIGINT	8	-	-
BINARY	-	254	-	-	-	-	CHAR	-	-	S
BINARY	255	32672	-	-	-	-	VARCHAR	-	-	S
BINARY	32673	2147483647	-	-	-	-	BLOB	2147483647	-	-
BIT	-	-	-	-	-	-	SMALLINT	2	-	-
BLOB	-	-	-	-	-	-	BLOB	2147483647	-	-
BOOLEAN	-	-	-	-	-	-	SMALLINT	2	-	-
CHAR	-	254	-	-	-	-	CHAR	-	-	-
CHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
CHAR	32673	2147483647	-	-	-	-	CLOB	2147483647	-	-
CLOB	-	-	-	-	-	-	CLOB	2147483647	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-
DATE	-	-	-	-	-	-	TIMESTAMP ¹	-	-	-
DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
DECIMAL	32	38	0	38	-	-	DOUBLE	8	-	-
DOUBLE	-	-	-	-	-	-	DOUBLE	8	-	-
FLOAT	-	-	-	-	-	-	FLOAT	4	-	-
INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
LONGVARCHAR	-	32672	-	-	-	-	VARCHAR	-	-	-
LONGVARCHAR	32673	2147483647	-	-	-	-	CLOB	2147483647	-	-
LONGVARBINARY	-	32672	-	-	-	-	VARCHAR	-	-	S
LONGVARBINARY	32673	2147483647	-	-	-	-	BLOB	2147483647	-	-
LONGNVARCHAR	-	16336	-	-	-	-	VARGRAPHIC	-	-	-
LONGNVARCHAR ²	16337	1073741823	-	-	-	-	DBCLOB	-	-	-
NCHAR ²	-	127	-	-	-	-	GRAPHIC	-	-	-
NCHAR ²	128	16336	-	-	-	-	VARGRAPHIC	-	-	-
NCHAR ²	16337	1073741823	-	-	-	-	DBCLOB	-	-	-
NCLOB ²	-	-	-	-	-	-	DBCLOB	-	-	-
NUMERIC	1	31	0	31	-	-	DECIMAL	-	-	-
NUMERIC	32	38	0	38	-	-	DOUBLE	8	-	-
NVARCHAR ²	-	16336	-	-	-	-	VARGRAPHIC	-	-	-
NVARCHAR ²	16337	1073741823	-	-	-	-	DBCLOB	-	-	-
REAL	-	-	-	-	-	-	REAL	4	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
TIME	-	-	-	-	-	-	TIME	3	-	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP(6)	10	6	-
TIMESTAMP(p)	-	-	-	-	-	-	TIMESTAMP(6)	10	6	-
TINYINT	-	-	-	-	-	-	SMALLINT	2	-	-
VARBINARY	-	32672	-	-	-	-	VARCHAR	-	-	S
VARBINARY	32673	2147483647	-	-	-	-	BLOB	2147483647	-	-
VARCHAR	-	32672	-	-	-	-	VARCHAR	-	-	-
VARCHAR	32673	2147483647	-	-	-	-	CLOB	2147483647	-	-

Tabela 114. Mapeamentos de Tipos de Dados de Encaminhamento Padrão JDBC (continuação)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
---------------------	----------------------	----------------------	------------------------	------------------------	-----------------------	-----------------------------	-----------------------	----------------------	-----------------	-------------------------

Nota:

1. O tipo federado é `TIMESTAMP(0)` se o parâmetro de configuração `date_compat` estiver definido como `ON`.
2. Os tipos de dados que são suportados apenas pelo driver JDBC 4.0: `NCHAR`, `NVARCHAR`, `LONGVARCHAR` e `NCLOB`.

Os seguintes tipos de dados não são suportados pelo wrapper JDBC: `DATALINK`, `OTHER`, `JAVA_OBJECT`, `DISTINCT`, `STRUCT`, `ARRAY` e `REF`.

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Microsoft SQL Server

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do Microsoft SQL Server.

Tabela 115. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Microsoft SQL Server

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
bigint	-	-	-	-	-	-	BIGINT	-	-	-
binary	1	254	-	-	-	-	CHARACTER	-	-	S
binary	255	8000	-	-	-	-	VARCHAR	-	-	S
bit	-	-	-	-	-	-	SMALLINT	2	-	-
char	1	254	-	-	-	-	CHAR	-	-	N
char	255	8000	-	-	-	-	VARCHAR	-	-	N
datetime	-	-	-	-	-	-	TIMESTAMP(6)	10	6	-
decimal	1	31	0	31	-	-	DECIMAL	-	-	-
decimal	32	38	0	38	-	-	DOUBLE	-	-	-
float	-	8	-	-	-	-	DOUBLE	8	-	-
float	-	4	-	-	-	-	REAL	4	-	-
image	-	-	-	-	-	-	BLOB	2147483647	-	S
int	-	-	-	-	-	-	INTEGER	4	-	-
money	-	-	-	-	-	-	DECIMAL	19	4	-
nchar	1	127	-	-	-	-	CHAR	-	-	N
nchar	128	4000	-	-	-	-	VARCHAR	-	-	N
numeric	1	31	0	31	-	-	DECIMAL	-	-	-
numeric	32	38	0	38	-	-	DOUBLE	8	-	-
ntext	-	-	-	-	-	-	CLOB	2147483647	-	S
nvarchar	1	4000	-	-	-	-	VARCHAR	-	-	N
real	-	-	-	-	-	-	REAL	4	-	-
smallint	-	-	-	-	-	-	SMALLINT	2	-	-
smalldatetime	-	-	-	-	-	-	TIMESTAMP(6)	10	6	-
smallmoney	-	-	-	-	-	-	DECIMAL	10	4	-
SQL_BIGINT	-	-	-	-	-	-	BIGINT	-	-	-
SQL_BINARY	1	254	-	-	-	-	CHARACTER	-	-	S
SQL_BINARY	255	8000	-	-	-	-	VARCHAR	-	-	S
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-

Tabela 115. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Microsoft SQL Server (continuação)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N
SQL_CHAR	255	8000	-	-	-	-	VARCHAR	-	-	N
SQL_DATE	-	-	-	-	-	-	DATE	4	-	-
SQL_DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_DECIMAL	32	38	0	38	-	-	DOUBLE	8	-	-
SQL_DOUBLE	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_GUID	-	-	-	-	-	-	VARCHAR	-	-	S
SQL_INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
SQL_LONGVARCHAR	-	-	-	-	-	-	CLOB	2147483647	-	N
SQL_LONGVARBINARY	-	-	-	-	-	-	BLOB	-	-	S
SQL_NUMERIC	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_NUMERIC	32	38	0	38	-	-	DOUBLE	8	-	-
SQL_REAL	-	-	-	-	-	-	REAL	8	-	-
SQL_SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_TIME	-	-	-	-	-	-	TIME	3	-	-
SQL_TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	10	6	-
SQL_TINYINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_VARBINARY	1	8000	-	-	-	-	VARCHAR	-	-	S
SQL_VARCHAR	1	8000	-	-	-	-	VARCHAR	-	-	N
SQL_WCHAR	1	254	-	-	-	-	CHARACTER	-	-	N
SQL_WCHAR	255	8800	-	-	-	-	VARCHAR	-	-	N
SQL_WLONGVARCHAR	-	1073741823	-	-	-	-	CLOB	2147483647	-	N
SQL_WVARCHAR	1	16336	-	-	-	-	VARCHAR	-	-	N
text	-	-	-	-	-	-	CLOB	-	-	N
timestamp	-	-	-	-	-	-	VARCHAR	8	-	S
tinyint	-	-	-	-	-	-	SMALLINT	2	-	-
uniqueidentifier	1	4000	-	-	S	-	VARCHAR	16	-	S
varbinary	1	8000	-	-	-	-	VARCHAR	-	-	S
varchar	1	8000	-	-	-	-	VARCHAR	-	-	N

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do ODBC

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do ODBC.

Tabela 116. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para ODBC (Não são mostradas todas as colunas)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
SQL_BIGINT	-	-	-	-	-	-	BIGINT	8	-	-

Tabela 116. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para ODBC (Não são mostradas todas as colunas) (continuação)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
SQL_BINARY	1	254	-	-	-	-	CHARACTER	-	-	S
SQL_BINARY	255	32672	-	-	-	-	VARCHAR	-	-	S
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N
SQL_CHAR	255	32672	-	-	-	-	VARCHAR	-	-	N
SQL_DATE	-	-	-	-	-	-	DATE	-	-	-
SQL_DATE	-	-	-	-	-	-	TIMESTAMP ¹	-	-	-
SQL_DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_DECIMAL	32	38	0	38	-	-	DOUBLE	8	-	-
SQL_DOUBLE	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_FLOAT	-	8	-	-	-	-	FLOAT	8	-	-
SQL_FLOAT	-	4	-	-	-	-	FLOAT	4	-	-
SQL_INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
SQL_LONGVARCHAR	-	-	-	-	-	-	CLOB	2147483647	-	N
SQL_LONGVARBINARY	-	-	-	-	-	-	BLOB	2147483647	-	S
SQL_NUMERIC	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_NUMERIC	32	32	0	31	-	-	DOUBLE	8	-	-
SQL_REAL	-	-	-	-	-	-	REAL	4	-	-
SQL_SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_TIMESTAMP	-	-	-	-	-	-	TIMESTAMP(6)	10	6	-
SQL_TIMESTAMP(p)	-	-	-	-	-	-	TIMESTAMP(6)	10	6	-
SQL_TYPE_DATE	-	-	-	-	-	-	DATE	4	-	-
SQL_TYPE_TIME	-	-	-	-	-	-	TIME	3	-	-
SQL_TYPE_TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	10	-	-
SQL_TINYINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_VARBINARY	1	32672	-	-	-	-	VARCHAR	-	-	S
SQL_VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	N
SQL_WCHAR	1	127	-	-	-	-	CHAR	-	-	N
SQL_WCHAR	128	16336	-	-	-	-	VARCHAR	-	-	N
SQL_WVARCHAR	1	16336	-	-	-	-	VARCHAR	-	-	N
SQL_WLONGVARCHAR	-	1073741823	-	-	-	-	CLOB	2147483647	-	N

Nota:

1. O tipo federado é TIMESTAMP(0) se o parâmetro de configuração date_compat estiver definido como ON.

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Oracle NET8

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do Oracle NET8.

Tabela 117. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Oracle NET8

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
BLOB	0	0	0	0	-	\0	BLOB	2147483647	0	S
CHAR	1	254	0	0	-	\0	CHAR	0	0	N
CHAR	255	2000	0	0	-	\0	VARCHAR	0	0	N
CLOB	0	0	0	0	-	\0	CLOB	2147483647	0	N
DATE	0	0	0	0	-	\0	TIMESTAMP(6)	0	0	N
FLOAT	1	126	0	0	-	\0	DOUBLE	0	0	N
LONG	0	0	0	0	-	\0	CLOB	2147483647	0	N
LONG RAW	0	0	0	0	-	\0	BLOB	2147483647	0	S
NUMBER	10	18	0	0	-	\0	BIGINT	0	0	N
NUMBER	1	38	-84	127	-	\0	DOUBLE	0	0	N
NUMBER	1	31	0	31	-	>=	DECIMAL	0	0	N
NUMBER	1	4	0	0	-	\0	SMALLINT	0	0	N
NUMBER	5	9	0	0	-	\0	INTEGER	0	0	N
NUMBER	-	10	0	0	-	\0	DECIMAL	0	0	N
RAW	1	2000	0	0	-	\0	VARCHAR	0	0	S
ROWID	0	0	0	NULL	-	\0	CHAR	18	0	N
TIMESTAMP(p) ¹	-	-	-	-	-	\0	TIMESTAMP(6)	10	6	N
VARCHAR2	1	4000	0	0	-	\0	VARCHAR	0	0	N

Nota:

1.

- **TIMESTAMP(p)** representa um registro de data e hora com escala variável de 0-9. A escala do registro de data e hora do Oracle é mapeada para **TIMESTAMP(6)** por padrão. É possível alterar esse mapeamento de tipo padrão e mapear o **TIMESTAMP** do Oracle para um **TIMESTAMP** federado da mesma escala, utilizando um mapeamento de tipo definido pelo usuário.
- Esse mapeamento de tipo é válido apenas para as configurações de cliente e servidor do Oracle 9i (ou posterior).

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Sybase

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do Sybase.

Tabela 118. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Sybase CTLIB

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
binary	1	254	-	-	-	-	CHAR	-	-	S
binary	255	32672	-	-	-	-	VARCHAR	-	-	S
bit	-	-	-	-	-	-	SMALLINT	-	-	-
char	1	254	-	-	-	-	CHAR	-	-	N
char	255	32672	-	-	-	-	VARCHAR	-	-	N
char null (consulte varchar)										
date	-	-	-	-	-	-	DATE	-	-	-
date	-	-	-	-	-	-	TIMESTAMP ¹	-	-	-
datetime	-	-	-	-	-	-	TIMESTAMP(6)	-	-	-

Tabela 118. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Sybase CTLIB (continuação)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
datetimn	-	-	-	-	-	-	TIMESTAMP	-	-	-
decimal	1	31	0	31	-	-	DECIMAL	-	-	-
decimal	32	38	0	38	-	-	DOUBLE	-	-	-
decimaln	1	31	0	31	-	-	DECIMAL	-	-	-
decimaln	32	38	0	38	-	-	DOUBLE	-	-	-
float	-	4	-	-	-	-	REAL	-	-	-
float	-	8	-	-	-	-	DOUBLE	-	-	-
floatn	-	4	-	-	-	-	REAL	-	-	-
floatn	-	8	-	-	-	-	DOUBLE	-	-	-
image	-	-	-	-	-	-	BLOB	-	-	-
int	-	-	-	-	-	-	INTEGER	-	-	-
intn	-	-	-	-	-	-	INTEGER	-	-	-
money	-	-	-	-	-	-	DECIMAL	19	4	-
moneyn	-	-	-	-	-	-	DECIMAL	19	4	-
nchar	1	254	-	-	-	-	CHAR	-	-	N
nchar	255	32672	-	-	-	-	VARCHAR	-	-	N
nchar null (consulte nvarchar)										
numeric	1	31	0	31	-	-	DECIMAL	-	-	-
numeric	32	38	0	38	-	-	DOUBLE	-	-	-
numericn	1	31	0	31	-	-	DECIMAL	-	-	-
numericn	32	38	0	38	-	-	DOUBLE	-	-	-
nvarchar	1	32672	-	-	-	-	VARCHAR	-	-	N
real	-	-	-	-	-	-	REAL	-	-	-
smalldatetime	-	-	-	-	-	-	TIMESTAMP(6)	-	-	-
smallint	-	-	-	-	-	-	SMALLINT	-	-	-
smallmoney	-	-	-	-	-	-	DECIMAL	10	4	-
sysname	-	-	-	-	-	-	VARCHAR	30	-	N
text	-	-	-	-	-	-	CLOB	-	-	-
time	-	-	-	-	-	-	TIME	-	-	-
timestamp	-	-	-	-	-	-	VARCHAR	8	-	S
tinyint	-	-	-	-	-	-	SMALLINT	-	-	-
unichar ²	1	254	-	-	-	-	CHAR	-	-	N
unichar ²	255	32672	-	-	-	-	VARCHAR	-	-	N
unichar null (consulte univarchar)										
univarchar ²	1	32672	-	-	-	-	VARCHAR	-	-	N
varbinary	1	32672	-	-	-	-	VARCHAR	-	-	S
varchar	1	32672	-	-	-	-	VARCHAR	-	-	N

Nota:

1. O tipo federado é TIMESTAMP(0) se o parâmetro de configuração date_compat estiver definido como ON.
2. Válido para bancos de dados federados não-Unicode.

Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Origens de Dados do Teradata

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do Teradata.

Tabela 119. Mapeamentos de Tipos de Dados de Encaminhamento Padrão para Teradata (Não são mostradas todas as colunas)

Nome de tipo remoto	Comp inferior remoto	Comp superior remoto	Escala inferior remota	Escala superior remota	Dados de bits remotos	Operadores de dados remotos	Nome de tipo federado	Comprimento Federado	Escala federada	Dados de bits federados
BIGINT ¹	-	-	-	-	-	-	BIGINT ¹	-	-	-
BLOB	1	2097088000	-	-	-	-	BLOB	-	-	-
BYTE	1	254	-	-	-	-	CHAR	-	-	S
BYTE	255	32672	-	-	-	-	VARCHAR	-	-	S
BYTE	32673	64000	-	-	-	-	BLOB	-	-	-
BYTEINT	-	-	-	-	-	-	SMALLINT	-	-	-
CHAR	1	254	-	-	-	-	CHARACTER	-	-	-
CHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
CHAR	32673	64000	-	-	-	-	CLOB	-	-	-
CLOB	1	2097088000 (Latin)	-	-	-	-	CLOB	-	-	-
CLOB	1	1048544000 (Unicode)	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-
DATE	-	-	-	-	-	-	TIMESTAMP ²	-	-	-
DECIMAL	1	31 ³	0	31 ³	-	-	DECIMAL	-	-	-
DOUBLE PRECISION	-	-	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	-	-
GRAPHIC	128	16336	-	-	-	-	VARGRAPHIC	-	-	-
GRAPHIC	16337	32000	-	-	-	-	DBCLOB	-	-	-
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
INTERVAL	-	-	-	-	-	-	CHAR	-	-	-
NUMERIC	1	18	0	18	-	-	DECIMAL	-	-	-
REAL	-	-	-	-	-	-	DOUBLE	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIME	0	21	0	21	-	-	TIME	-	-	-
TIMESTAMP(<i>p</i>)	-	-	<i>p</i>	<i>p</i>	-	-	TIMESTAMP(6) ¹⁰	10	6	-
VARBYTE	1	32762	-	-	-	-	VARCHAR	-	-	S
VARBYTE	32763	64000	-	-	-	-	BLOB	-	-	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
VARCHAR	32673	64000	-	-	-	-	CLOB	-	-	-
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	-	-
VARGRAPHIC	16337	32000	-	-	-	-	DBCLOB	-	-	-

Notas:

1. The BIGINT data type applies to Teradata version 2 release 6 or later.
2. O tipo federado é TIMESTAMP(0) se o parâmetro de configuração date_compat estiver definido como ON.
3. The remote upper length and remote upper scale value 31 applies to the Teradata version 2 release 6 or later. For earlier versions, this value is 18.

Mapeamentos de Tipos de Dados de Encaminhamento de Amostra

É possível utilizar os mapeamentos de tipos de encaminhamento de amostra para aproveitar as vantagens de suporte para o tipo de dado `TIMESTAMP` com uma precisão.

Para as origens de dados do Informix, esses mapeamentos de tipos são utilizados para os tipos de colunas de apelidos, parâmetros de procedimentos federados, conjuntos de resultados dos procedimentos intermediários e federados.

Para origens de dados diferentes do Informix, esses mapeamentos de tipos afetam apenas os mapeamentos para os tipos de colunas de apelidos e parâmetros de procedimentos federados. Os mapeamentos não afetam os conjuntos de resultados de procedimentos intermediários e federados.

Mapeamentos de Tipos de Dados de Encaminhamento - Exemplo do Informix

Ao criar objetos federados, é possível utilizar o mapeamento de tipo de encaminhamento de amostra para Informix.

É necessário criar esses mapeamentos antes de criar um objeto federado.

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(0)
  TO SERVER TYPE informix REMOTE TYPE datetime(0,10);
```

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(1)
  TO SERVER TYPE informix REMOTE TYPE datetime(0,11);
```

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(2)
  TO SERVER TYPE informix REMOTE TYPE datetime(0,12);
```

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(3)
  TO SERVER TYPE informix REMOTE TYPE datetime(0,13);
```

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(4)
  TO SERVER TYPE informix REMOTE TYPE datetime(0,14);
```

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(5)
  TO SERVER TYPE informix REMOTE TYPE datetime(0,15);
```

Mapeamentos de Tipos de Dados de Encaminhamento - Exemplo do Microsoft SQL Server

Ao criar objetos federados, é possível utilizar o mapeamento de tipo de encaminhamento de amostra para Microsoft SQL Server.

É necessário criar esses mapeamentos antes de criar um apelido ou procedimento federado.

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(3)
  TO SERVER TYPE mssqlserver REMOTE TYPE "datetime";
```

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(0)
  TO SERVER TYPE mssqlserver REMOTE TYPE "datetime";
```

Mapeamentos de Tipos de Dados de Encaminhamento - Exemplo do Oracle

Ao criar objetos federados, é possível utilizar o mapeamento de tipo de encaminhamento de amostra para Oracle.

É necessário criar esses mapeamentos antes de criar um apelido ou procedimento federado.

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(0)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(0);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(1)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(1);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(2)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(2);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(3)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(3);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(4)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(4);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(5)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(5);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(7)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(7);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(8)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(8);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(9)
  TO SERVER TYPE oracle REMOTE TYPE timestamp(9);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(0)
  TO SERVER TYPE oracle REMOTE TYPE date;
```

Mapeamentos de Tipos de Dados de Encaminhamento - Exemplo do Sybase

Ao criar objetos federados, é possível utilizar o mapeamento de tipo de encaminhamento de amostra para Sybase.

É necessário criar esses mapeamentos antes de criar um apelido ou procedimento federado.

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(3)
  TO SERVER TYPE sybase REMOTE TYPE datetime);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(0)
  TO SERVER TYPE sybase REMOTE TYPE smalldatetime);
```

Mapeamentos de Tipos de Dados de Encaminhamento - Exemplo do Teradata

Ao criar objetos federados, é possível utilizar o mapeamento de tipo de encaminhamento de amostra para Teradata.

É necessário criar esses mapeamentos antes de criar um apelido ou procedimento federado.

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(0)
  TO SERVER TYPE teradata REMOTE TYPE timestamp(0);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(1)
  TO SERVER TYPE teradata REMOTE TYPE timestamp(1);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(2)
  TO SERVER TYPE teradata REMOTE TYPE timestamp(2);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(3)
  TO SERVER TYPE teradata REMOTE TYPE timestamp(3);
```

```
CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(4)
  TO SERVER TYPE teradata REMOTE TYPE timestamp(4);

CREATE TYPE MAPPING FROM LOCAL TYPE timestamp(5)
  TO SERVER TYPE teradata REMOTE TYPE timestamp(5);
```

Mapeamentos de tipo de dados reversos padrão

Para a maioria das origens de dados, os mapeamentos de tipo padrão estão nos wrappers.

Os dois tipos de mapeamentos entre os tipos de dados de origem de dados e os tipos de dados do banco de dados federado são os mapeamentos de tipos de encaminhamento e os mapeamentos de tipos reversos. Em um mapeamento de tipo de encaminhamento, o mapeamento é de um tipo remoto a um tipo local comparável. O outro tipo de mapeamento é um mapeamento de tipo reverso, utilizado com o DDL transparente para criar ou modificar tabelas remotas.

Os mapeamentos de tipos padrão para as origens de dados da família DB2 estão no wrapper DRDA. Os mapeamentos de tipos padrão para Informix estão no wrapper INFORMIX, e assim por diante.

Ao definir uma tabela ou visualização remota para o banco de dados federado, a definição inclui um mapeamento de tipo reverso. O mapeamento é de um banco de dados federado local para cada coluna e o tipo de dado remoto correspondente. Por exemplo, há um mapeamento de tipo reverso padrão no qual o tipo real REAL aponta para o tipo do Informix SMALLFLOAT.

Os bancos de dados federados não suportam mapeamentos para LONG VARCHAR, LONG VARCHARIC e os tipos definidos pelo usuário.

Ao utilizar a instrução CREATE TABLE para criar uma tabela remota, você especifica os tipos de dados locais que deseja incluir na tabela remota. Esses mapeamentos de tipos reversos padrão designarão os tipos remotos correspondentes a essas colunas. Por exemplo, suponha que você utilize a instrução CREATE TABLE para definir uma tabela Informix com uma coluna C2. Você especifica BIGINT como o tipo de dado para C2 na instrução. O mapeamento de tipo reverso padrão de BIGINT depende de em qual versão do Informix a tabela está sendo criada. O mapeamento para C2 na tabela Informix será ao DECIMAL no Informix Versão 8 e INT8 no Informix Versão 9.

É possível substituir um mapeamento de tipo reverso padrão ou criar um novo mapeamento de tipo reverso com a instrução CREATE TYPE MAPPING.

As tabelas a seguir mostram os mapeamentos reversos padrão entre os tipos de dados locais do banco de dados federado e os tipos de dados da origem de dados remota.

Esses mapeamentos são válidos com todas as versões suportadas, a menos que seja observado de outra maneira.

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Banco de Dados DB2 para Linux, UNIX, e Windows

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para origens de dados do banco de Dados DB2 para Linux, UNIX, e Windows.

Tabela 120. Mapeamentos de Tipos de Dados Reversos Padrão para Banco de Dados DB2 para Linux, UNIX, e Windows (Não são mostradas todas as colunas)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits federados
BIGINT	-	8	-	-	-	-	BIGINT	-	-	-
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHAR	-	-	N
CHARACTER	-	-	-	-	S	-	CHAR	-	-	S
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE ¹	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DECFLOAT ²	-	8	-	-	-	-	DECFLOAT	-	0	-
DECFLOAT ²	-	16	-	-	-	-	DECFLOAT	-	0	-
DOUBLE	-	8	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	8	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	-	-	-	-	-	REAL	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP(<i>p</i>)	-	-	<i>p</i>	<i>p</i>	-	-	TIMESTAMP(<i>p</i>)	-	<i>p</i> ³	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	S	-	VARCHAR	-	-	S
VARGRAPH	-	-	-	-	-	-	VARGRAPHIC	-	-	N
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	-

Nota:

1. Quando o parâmetro `date_compat` é configurado como OFF, o DATE federado mapeia para TIMESTAMP(0).
2. A opção do servidor `SAME_DECFLT_ROUNDING` é configurada como N por padrão e as operações não serão rebaixasadas à origem de dados remota, a menos que `SAME_DECFLT_ROUNDING` seja configurado como Y. Para obter informações sobre a opção do servidor `SAME_DECFLT_ROUNDING`, consulte as Referência das Opções do Banco de Dados DB2.
3. Para a Versão 9.5 ou anterior, a escala remota para TIMESTAMP é 6.

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do DB2 para System i

A tabela a seguir lista os mapeamentos de tipo de dados reversos padrão para origens de dados do DB2 para System i.

Tabela 121. Mapeamentos de Tipos de Dados Reversos Padrão para DB2 para System i (Não são mostradas todas as colunas)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHARACTER	-	-	N
CHARACTER	-	-	-	-	S	-	CHARACTER	-	-	S
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	NUMERIC	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	FLOAT	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP(p)-	-	-	p	p	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	S	-	VARCHAR	-	-	S
VARGRAPHIC	-	-	-	-	-	-	VARG	-	-	N

Mapeamentos de Tipos de Dados Reversos Padrão para origens de dados do DB2 para VM e VSE

A tabela a seguir lista o tipo de dados reversos padrão para origens de dados do DB2 para VM e VSE.

Tabela 122. Mapeamentos de Tipos de Dados Reversos Padrão para DB2 Server para VM e VSE (Não são mostradas todas as colunas)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHAR	-	-	-
CHARACTER	-	-	-	-	S	-	CHAR	-	-	S
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	REAL	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP(p)-	-	-	p	p	-	-	TIMESTAMP	-	-	-

Tabela 122. Mapeamentos de Tipos de Dados Reversos Padrão para DB2 Server para VM e VSE (Não são mostradas todas as colunas) (continuação)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	-
VARCHAR	-	-	-	-	S	-	VARCHAR	-	-	S
VARGRAPH	-	-	-	-	-	-	VARGRAPH	-	-	N

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do DB2 para z/OS

A tabela a seguir lista o tipo de dados reversos padrão para origens de dados do DB2 para z/OS.

Tabela 123. Mapeamentos de Tipos de Dados Reversos Padrão para DB2 para z/OS (Não são mostradas todas as colunas)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT	0	8	0	0	" "	"/0"	BIGINT	0	0	N
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHAR	-	-	N
CHARACTER	-	-	-	-	S	-	CHAR	-	-	S
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	8	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	REAL	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP(<i>p</i>)	-	-	<i>p</i>	<i>p</i>	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	S	-	VARCHAR	-	-	S
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	N

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Informix

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do Informix.

Tabela 124. Mapeamentos de Tipos de Dados Reversos Padrão do Informix

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT ¹	-	-	-	-	-	-	DECIMAL	19	-	-

Tabela 124. Mapeamentos de Tipos de Dados Reversos Padrão do Informix (continuação)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT ²	-	-	-	-	-	-	INT8	-	-	-
BLOB	1	2147483647	-	-	-	-	BYTE	-	-	-
CHARACTER	-	-	-	-	N	-	CHAR	-	-	-
CHARACTER	-	-	-	-	S	-	BYTE	-	-	-
CLOB	1	2147483647	-	-	-	-	TEXT	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	SMALLFLOAT	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	DATETIME	6	10	-
TIMESTAMP	-	10	-	-	-	-	DATETIME	0	15	-
VARCHAR	1	254	-	-	N	-	VARCHAR	-	-	-
VARCHAR ¹	255	32672	-	-	N	-	TEXT	-	-	-
VARCHAR	-	-	-	-	S	-	BYTE	-	-	-
VARCHAR ²	255	2048	-	-	N	-	LVARCHAR	-	-	-
VARCHAR ²	2049	32672	-	-	N	-	TEXT	-	-	-

Nota:

1. Este mapeamento de tipo é válido apenas com o servidor Informix Versão 8 (ou inferior).
2. Este mapeamento de tipo é válido apenas com o servidor Informix Versão 9 (ou superior).

Para os tipos de dados Informix DATETIME, o servidor federado utiliza o qualificador de alto nível do Informix, como REMOTE_LENGTH, e o qualificador de baixo nível do Informix, como REMOTE_SCALE.

Os qualificadores do Informix são as constantes "TU_" definidas no arquivo Informix Client SDK datatime.h. As constantes são:

0 = YEAR	8 = MINUTE	13 = FRACTION(3)
2 = MONTH	10 = SECOND	14 = FRACTION(4)
4 = DAY	11 = FRACTION(1)	15 = FRACTION(5)
6 = HOUR	12 = FRACTION(2)	

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do JDBC

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do JDBC, que estão em conformidade com os mapeamentos de tipos do driver JDBC do DB2.

Tabela 125. Mapeamentos de Tipos de Dados Reversos Padrão JDBC

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT	-	-	-	-	-	-	BIGINT	-	-	-
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	-	-	-	-	S	-	BINARY	-	-	-
CHAR	-	-	-	-	-	-	CHAR	-	-	-
CLOB	-	-	-	-	-	-	CLOB	-	-	-

Tabela 125. Mapeamentos de Tipos de Dados Reversos Padrão JDBC (continuação)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	NCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	NCHAR	-	-	-
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	REAL	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	-	-
TIMESTAMP(<i>p</i>)	-	-	<i>p</i>	<i>p</i>	-	-	TIMESTAMP(<i>p</i>)	-	min(9, <i>p</i>)	-
VARCHAR	-	-	-	-	S	-	VARBINARY	-	-	-
VARCHAR	-	-	-	-	N	-	VARCHAR	-	-	-
VARGRAPHIC	-	-	-	-	-	-	NVARCHAR	-	-	-

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Microsoft SQL Server

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do Microsoft SQL Server.

Tabela 126. Mapeamentos de Tipos de Dados Reversos Padrão para Microsoft (Não são mostradas todas as colunas)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT	-	-	-	-	-	-	bigint	-	-	-
BLOB	-	-	-	-	-	-	image	-	-	-
CHARACTER	-	-	-	-	S	-	binary	-	-	-
CHARACTER	-	-	-	-	N	-	char	-	-	-
CLOB	-	-	-	-	-	-	text	-	-	-
DATE	-	4	-	-	-	-	datetime	-	-	-
DECIMAL	-	-	-	-	-	-	decimal	-	-	-
DOUBLE	-	8	-	-	-	-	float	-	-	-
INTEGER	-	-	-	-	-	-	int	-	-	-
SMALLINT	-	-	-	-	-	-	smallint	-	-	-
REAL	-	4	-	-	-	-	real	-	-	-
TIME	-	3	-	-	-	-	datetime	-	-	-
TIMESTAMP	-	10	-	-	-	-	datetime	-	-	-
VARCHAR	1	8000	-	-	N	-	varchar	-	-	-
VARCHAR	8001	32672	-	-	N	-	text	-	-	-
VARCHAR	1	8000	-	-	S	-	varbinary	-	-	-
VARCHAR	8001	32672	-	-	S	-	image	-	-	-

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do ODBC

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do ODBC.

Tabela 127. Mapeamentos de Tipos de Dados Reversos Padrão para ODBC (Não são mostradas todas as colunas)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT	-	-	-	-	-	-	SQL_BIGINT	-	-	-
BLOB	-	-	-	-	-	-	SQL_LONGVARBINARY	-	-	-
CHAR	-	-	-	-	S	-	SQL_BINARY	-	-	-
CHAR	-	-	-	-	N	-	SQL_CHAR	-	-	-
CLOB	-	-	-	-	-	-	SQL_LONGVARCHAR	-	-	-
DATE	-	4	-	-	-	-	SQL_TYPE_DATE	-	-	-
DBCLOB	-	-	-	-	-	-	SQL_WLONGVARCHAR	-	-	-
DECIMAL	-	-	-	-	-	-	SQL_DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	SQL_DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	SQL_FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	SQL_WCHAR	-	-	-
INTEGER	-	-	-	-	-	-	SQL_INTEGER	-	-	-
REAL	-	4	-	-	-	-	SQL_REAL	-	-	-
SMALLINT	-	-	-	-	-	-	SQL_SMALLINT	-	-	-
TIME	-	3	-	-	-	-	SQL_TYPE_TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	SQL_TYPE_TIMESTAMP(p)	-	-	-
VARCHAR	-	-	-	-	S	-	SQL_VARBINARY	-	-	-
VARCHAR	-	-	-	-	N	-	SQL_VARCHAR	-	-	-
VARGRAPHIC	-	-	-	-	S	-	SQL_WVARCHAR	-	-	-

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Oracle NET8

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do Oracle NET8.

Tabela 128. Mapeamentos de Tipos de Dados Reversos Padrão do Oracle NET8

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT	0	8	0	0	N	\0	NUMBER	19	0	N
BLOB	0	2147483647	0	0	S	\0	BLOB	0	0	S
CHARACTER	1	254	0	0	N	\0	CHAR	0	0	N
CHARACTER	1	254	0	0	S	\0	RAW	0	0	S
CLOB	0	2147483647	0	0	N	\0	CLOB	0	0	N
DATE ¹	0	4	0	0	N	\0	DATE	0	0	N
DECIMAL	0	0	0	0	N	\0	NUMBER	0	0	N
DECFLOAT	0	8	0	0	N	\0	NUMBER	0	0	N
DECFLOAT	0	16	0	0	N	\0	NUMBER	0	0	N
DOUBLE	0	8	0	0	N	\0	FLOAT	126	0	N
FLOAT	0	8	0	0	N	\0	FLOAT	126	0	N

Tabela 128. Mapeamentos de Tipos de Dados Reversos Padrão do Oracle NET8 (continuação)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
INTEGER	0	4	0	0	N	\0	NUMBER	10	0	N
REAL	0	4	0	0	N	\0	FLOAT	63	0	N
SMALLINT	0	2	0	0	N	\0	NUMBER	5	0	N
TIME	0	3	0	0	N	\0	DATE	0	0	N
TIMESTAMP ²	0	10	0	0	N	\0	DATE	0	0	N
TIMESTAMP(p) ³	-	-	-	-	N	\0	TIMESTAMP(p)	-	-	N
VARCHAR	1	4000	0	0	N	\0	VARCHAR2	0	0	N
VARCHAR	1	2000	0	0	S	\0	RAW	0	0	S

Nota:

- Quando o parâmetro `date_compat` é configurado como OFF, o DATE federado mapeia para a data do Oracle. Quando o parâmetro `date_compat` é configurado como ON, o DATE federado (equivalente ao TIMESTAMP(O)) mapeia para TIMESTAMP(O) do Oracle.
- Este mapeamento de tipo é válido apenas com o Oracle Versão 8.
- TIMESTAMP(p) representa um registro de data e hora com escala variável de 0-9 para o Oracle e 0-12 para a federação. Quando a escala é 0-9, o TIMESTAMP remoto do Oracle possui a mesma escala do TIMESTAMP federado. Se a escala do TIMESTAMP federado for maior que 9, a escala correspondente do TIMESTAMP do Oracle é 9, que é a escala mais alta do Oracle.
 - Este mapeamento de tipo é válido apenas com o Oracle Versão 9, 10 e 11.

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Sybase

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do Sybase.

Tabela 129. Mapeamentos de Tipos de Dados Reversos Padrão do Sybase CTLIB

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT	-	-	-	-	-	-	decimal	19	0	-
BLOB	-	-	-	-	-	-	image	-	-	-
CHARACTER	-	-	-	-	N	-	char	-	-	-
CHARACTER	-	-	-	-	S	-	binary	-	-	-
CLOB	-	-	-	-	-	-	text	-	-	-
DATE	-	-	-	-	-	-	datetime	-	-	-
DECIMAL	-	-	-	-	-	-	decimal	-	-	-
DOUBLE	-	-	-	-	-	-	float	-	-	-
INTEGER	-	-	-	-	-	-	número inteiro	-	-	-
REAL	-	-	-	-	-	-	real	-	-	-
SMALLINT	-	-	-	-	-	-	smallint	-	-	-
TIME	-	-	-	-	-	-	datetime	-	-	-
TIMESTAMP	-	-	-	-	-	-	datetime	-	-	-
VARCHAR ¹	1	255	-	-	N	-	varchar	-	-	-
VARCHAR ¹	256	32672	-	-	N	-	text	-	-	-
VARCHAR ²	1	16384	-	-	N	-	varchar	-	-	-

Tabela 129. Mapeamentos de Tipos de Dados Reversos Padrão do Sybase CTLIB (continuação)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
VARCHAR ²	16385	32672	-	-	N	-	text	-	-	-
VARCHAR ¹	1	255	-	-	S	-	varbinary	-	-	-
VARCHAR ¹	256	32672	-	-	S	-	image	-	-	-
VARCHAR ²	1	16384	-	-	S	-	varbinary	-	-	-
VARCHAR ²	16385	32672	-	-	S	-	image	-	-	-

Nota:

1. Este mapeamento de tipo é válido apenas para CTLIB com o servidor Sybase versão 12.0 (ou anterior).
2. Este mapeamento de tipo é válido apenas para CTLIB com o servidor Sybase versão 12.5 (ou posterior).

Mapeamentos de Tipos de Dados Reversos Padrão para Origens de Dados do Teradata

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do Teradata.

Tabela 130. Mapeamentos de Tipos de Dados Reversos Padrão para Teradata (Não são mostradas todas as colunas)

Nome de tipo federado	Comp inferior federado	Comp superior federado	Escala inferior federada	Escala superior federada	Dados de bits federados	Operadores de dados federados	Nome de tipo remoto	Comprimento remoto	Escala remota	Dados de bits remotos
BIGINT ¹	-	-	-	-	-	-	BIGINT	-	-	-
BLOB	1	2097088000	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHARACTER	-	-	-
CHARACTER	-	-	-	-	S	-	BYTE	-	-	-
CLOB	1	2097088000	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-
DBCLOB ²	1	64000	-	-	-	-	VARGRAPHIC	-	-	-
DECIMAL	1	18/31 ³	0	18/31 ³	-	-	DECIMAL	-	-	-
DECIMAL ⁴	19	31	0	31	-	-	FLOAT	8	-	-
DOUBLE	-	-	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	-
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
REAL	-	-	-	-	-	-	FLOAT	8	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIME	-	-	-	-	-	-	TIME	15	-	-
TIMESTAMP	10	10	6	6	-	-	TIMESTAMP	26	6	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	-
VARCHAR	-	-	-	-	S	-	VARBYTE	-	-	-
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	-

Notas:

1. The BIGINT data type applies to Teradata version 2 release 6 or later.
2. O tipo de dado VARGRAPHIC do Teradata pode conter apenas o comprimento especificado (1 a 32000) de um tipo de dado DBCLOB.
3. The value 31 applies to Teradata version 2 release 6 or later. The value 18 applies to earlier versions.
4. This DECIMAL row applies Teradata version 2 release 5 or earlier.

Mapeamentos de Tipos Padrão do Unicode

Especifique os mapeamentos de tipos de dados de encaminhamento de suporte das origens de dados e os mapeamentos de tipos de dados reversos para bancos de dados Unicode.

Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do JDBC

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do JDBC quando o banco de dados federado é um banco de dados Unicode.

Tabela 131. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do JDBC

UTF-8		JDBC
Tipo de dados	Tipo de dados	Comprimento
CHAR	CHAR	1 a 254 bytes
VARCHAR	VARCHAR	1 a 32672 bytes
CLOB	CLOB	-
GRAPHIC	NCHAR	1 a 127 caracteres
VARGRAPHIC	NVARCHAR	1 a 16336 caracteres
DBCLOB	NCLOB	-

Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do JDBC

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do JDBC quando o banco de dados federado é um banco de dados Unicode.

Tabela 132. Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do JDBC

UTF-8		JDBC
Tipo de dados	Comprimento	Tipo de dados
CHAR	1 a 254 bytes	CHAR
VARCHAR	1 a 32672 bytes	VARCHAR
CLOB	1 a 2147483647 bytes	CLOB
GRAPHIC	1 a 127 caracteres	NCHAR
VARGRAPHIC	1 a 16336 caracteres	NVARCHAR
DBCLOB	1 a 1073741823 caracteres	NCLOB

Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do Microsoft SQL Server

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do Microsoft SQL Server quando o banco de dados federado é um banco de dados Unicode.

Tabela 133. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do Microsoft SQL Server

UTF-8	Microsoft SQL Server	
Tipo de dados	Tipo de dados	Comprimento
CHAR	CHAR	1 a 254 bytes
VARCHAR	CHAR	255 a 8000 bytes
	VARCHAR	1 a 8000 bytes
CLOB	TEXT	-
GRAPHIC	NCHAR	1 a 127 caracteres
VARGRAPHIC	NCHAR	128 a 16336 caracteres
	NVARCHAR	1 a 16336 caracteres
DBCLOB	NTEXT	-

Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do Microsoft SQL Server

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do Microsoft SQL Server quando o banco de dados federado é um banco de dados Unicode.

Tabela 134. Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do Microsoft SQL Server

UTF-8	Microsoft SQL Server	
Tipo de dados	Comprimento	Tipo de dados
CHAR	1 a 254 bytes	CHAR
VARCHAR	1 a 32672 bytes	VARCHAR
CLOB	1 a 2 147 483 647 bytes	TEXT
GRAPHIC	1 a 127 caracteres	NCHAR
VARGRAPHIC	1 a 16336 caracteres	NVARCHAR
DBCLOB	1 a 1 073 741 823 caracteres	NTEXT

Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do NET8

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do NET8 quando o banco de dados federado é um banco de dados Unicode.

Tabela 135. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do NET8

UTF-8	Oracle	
Tipo de dados	Tipo de dados	Comprimento
CHAR	CHAR	1 a 254 bytes
VARCHAR	CHAR	255 a 2000 bytes
	VARCHAR2	1 a 4000 bytes
DBCLOB	NCLOB	
GRAPHIC	NCHAR	1 a 127 caracteres

Tabela 135. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do NET8 (continuação)

UTF-8	Oracle	
Tipo de dados	Tipo de dados	Comprimento
VARGRAPHIC	NCHAR	128 a 1000 caracteres
	NVARCHAR2	1 a 2000 caracteres

Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do NET8

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do NET8 quando o banco de dados federado é um banco de dados Unicode.

Tabela 136. Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do NET8

UTF-8	Oracle	
Tipo de dados	Comprimento	Tipo de dados
CHAR	1 a 254 bytes	CHAR
VARCHAR	1 a 4000 bytes	VARCHAR2
CLOB	1 a 2 147 483 647 bytes	CLOB
GRAPHIC	1 a 127 caracteres	NCHAR
VARGRAPHIC	1 a 2000 caracteres	NVARCHAR2
DBCLOB	1 a 1 073 741 823 caracteres	NCLOB

Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do ODBC

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do ODBC quando o banco de dados federado é um banco de dados Unicode.

Tabela 137. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do ODBC

UTF-8	ODBC	
Tipo de dados	Tipo de dados	Comprimento
CHAR	SQL_CHAR	1 a 254 bytes
VARCHAR	SQL_CHAR	255 a 32672 bytes
	SQL_VARCHAR	1 a 32672 bytes
CLOB	SQL_LONGVARCHAR	-
GRAPHIC	SQL_WCHAR	1 a 127 caracteres
VARGRAPHIC	SQL_WCHAR	128 a 16336 caracteres
	SQL_WVARCHAR	1 a 16336 caracteres
DBCLOB	SQL_WLONGVARCHAR	-

Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do ODBC

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do ODBC quando o banco de dados federado é um banco de dados Unicode.

Tabela 138. Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do ODBC

UTF-8		ODBC
Tipo de dados	Comprimento	Tipo de dados
CHAR	1 a 254 bytes	SQL_CHAR
VARCHAR	1 a 32672 bytes	SQL_VARCHAR
CLOB	1 a 2 147 483 647 bytes	SQL_LONGVARCHAR
GRAPHIC	1 a 127 caracteres	SQL_WCHAR
VARGRAPHIC	1 a 16336 caracteres	SQL_WVARCHAR
DBCLOB	1 a 1 073 741 823 caracteres	SQL_WLONGVARCHAR

Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do Sybase

A tabela a seguir lista os mapeamentos de tipos de dados de encaminhamento padrão para as origens de dados do Sybase CTLIB quando o banco de dados federado é um banco de dados Unicode.

Tabela 139. Mapeamentos de Tipos de Dados de Encaminhamento Padrão do Unicode para Origens de Dados do Sybase CTLIB

UTF-8	Sybase	
Tipo de dados	Tipo de dados	Comprimento
CHAR	char	1 a 254 bytes
	nchar	1 a 127 caracteres
VARCHAR	char	255 a 32672 bytes
	varchar	1 a 32672 bytes
	nchar	128 a 16336 caracteres
	nvarchar	1 a 16336 caracteres
CLOB	text	
GRAPHIC	unichar	1 a 127 caracteres
VARGRAPHIC	unichar	128 a 16336 caracteres
	univarchar	1 a 16336 caracteres

Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do Sybase

A tabela a seguir lista os mapeamentos de tipos de dados reversos padrão para as origens de dados do Sybase CTLIB quando o banco de dados federado é um banco de dados Unicode.

Tabela 140. Mapeamentos de Tipos de Dados Reversos Padrão do Unicode para Origens de Dados do Sybase CTLIB

UTF-8		Sybase
Tipo de dados	Comprimento	Tipo de dados
CHAR	1 a 254 bytes	char
VARCHAR	1 a 32672 bytes	varchar
CLOB	1 a 2 147 483 647 bytes	text
GRAPHIC	1 a 127 caracteres	unichar
VARGRAPHIC	1 a 16336 caracteres	univarchar

Tipos de Dados Suportados para Origens de Dados Não Relacionais

Para a maioria das origens de dados não relacionais, é necessário especificar as informações da coluna, incluindo o tipo de dados, ao criar os apelidos para acessar a origem de dados.

Alguns dos wrappers não relacionais criam todas as colunas necessárias para acessar uma origem de dados. Essas são denominadas *colunas fixas*. Outros wrappers permitem que você especifique alguns ou todos os tipos de dados para as colunas na instrução CREATE NICKNAME.

As seções a seguir listam os wrappers aos quais você pode especificar os tipos de dados e os tipos de dados que o wrapper suporta.

Tipos de Dados Suportados pelo Wrapper BioRS

A tabela a seguir lista os tipos de dados DB2 que o wrapper BioRS suporta.

Tabela 141. Tipos de Dados BioRS que São Mapeados aos Tipos de Dados DB2

Tipos de dados do BioRS	Tipos de dados do DB2
AUTHOR	CHARACTER, CLOB, VARCHAR
DATE	CHARACTER, CLOB, VARCHAR
NUMBER	CHARACTER, CLOB, VARCHAR
REFERENCE	CHARACTER, CLOB, VARCHAR
TEXT	CHARACTER, CLOB, VARCHAR

O comprimento máximo permitido para o tipo de dado CLOB é 5 megabytes.

Tipos de Dados Suportados pelo Wrapper Excel

A tabela a seguir lista os tipos de dados DB2 que o wrapper Excel suporta.

Tabela 142. Tipos de Dados Excel que São Mapeados aos Tipos de Dados DB2

Tipos de dados Excel	Tipos de dados do DB2
character	CHARACTER
date	DATE
number	DOUBLE
number	FLOAT

Tabela 142. Tipos de Dados Excel que São Mapeados aos Tipos de Dados DB2 (continuação)

Tipos de dados Excel	Tipos de dados do DB2
número inteiro	INTEGER
character	VARCHAR

Tipos de Dados Suportados pelo Wrapper Script

A tabela a seguir lista os tipos de dados DB2 que o wrapper Script suporta.

Tabela 143. Tipos de Dados Script que São Mapeados aos Tipos de Dados DB2

Tipos de dados XML	Tipos de dados do DB2
character	BLOB
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB (comprimento máximo é 5 megabytes)
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
número inteiro	INTEGER
number	REAL
número inteiro	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA

Tipos de Dados Suportados pelo Wrapper dos Arquivos Estruturados em Tabela

A tabela a seguir lista os tipos de dados DB2 que o wrapper dos arquivos estruturados em tabela suporta.

Tabela 144. Tipos de Dados dos Arquivos Estruturados em Tabela que São Mapeados aos Tipos de Dados DB2

Tipos de dados dos arquivos estruturados em tabelas	Tipos de dados do DB2
character	CHARACTER
character	CLOB (comprimento máximo é 5 megabytes)
number	DECIMAL
number	DOUBLE
number	FLOAT
número inteiro	INTEGER
number	REAL
número inteiro	SMALLINT
character	VARCHAR

Tipos de Dados Suportados pelo Wrapper de Serviços da Web

A tabela a seguir lista os tipos de dados DB2 que o wrapper de serviços da Web suporta. O wrapper de serviços da Web utiliza os tipos de dados XML.

Tabela 145. Tipos de Dados XML que São Mapeados aos Tipos de Dados DB2 para o Wrapper de Serviços da Web

Tipos de dados XML	Tipos de dados do DB2
character	BLOB
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB (comprimento máximo é 5 megabytes)
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
número inteiro	INTEGER
number	REAL
número inteiro	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA

Tipos de Dados Suportados pelo Wrapper XML

A tabela a seguir lista os tipos de dados DB2 que o wrapper XML suporta.

Tabela 146. Tipos de Dados XML que São Mapeados aos Tipos de Dados DB2 para o Wrapper XML

Tipos de dados XML	Tipos de dados do DB2
character	BLOB
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB (comprimento máximo é 5 megabytes)
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
número inteiro	INTEGER
number	REAL
número inteiro	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA
XML document	XML

Documentação Acessível

A documentação é fornecida em formato XHTML, que pode ser visualizado na maioria dos navegadores da Web.

O XHTML permite que você visualize a documentação de acordo com as preferências de exibição definidas no navegador. Ele permite também que você utilize leitores de tela e outras tecnologias assistidas.

Os diagramas de sintaxe são fornecidos em formato decimal com pontos. Esse formato estará disponível apenas se você acessar a documentação on-line utilizando um leitor de tela.

Avisos e Marcas Registradas

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

Avisos

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 ou outros direitos legalmente protegidos, poderá ser utilizado em substituição. 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 lhe garante direito algum 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:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 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 A ELAS NÃO SE LIMITANDO, AS 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.

Essas informações podem conter imprecisões técnicas ou erros tipográficos. São feitas alterações periódicas nas informações aqui contidas; tais alterações serão

incorporadas em futuras edições desta publicação. A IBM pode, a qualquer momento, aperfeiçoar 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. OOs 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:

Av. Pasteur 138-146
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 nesta publicação e todo o material licenciado disponível são fornecidos pela IBM sob os termos do Contrato com o Cliente IBM, do Contrato Internacional de Licença do Programa 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 em 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. Os 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 foram projetadas apenas com o propósito de planejamento. As informações aqui contidas estão sujeitas a alterações antes que os produtos descritos estejam disponíveis.

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.

LICENÇA DE COPYRIGHT:

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. Os programas de exemplo são fornecidos "NO ESTADO EM QUE SE ENCONTRAM", sem qualquer tipo de garantia. A IBM não deve ser responsabilizada por nenhum dano decorrente do uso dos programas de amostra.

Cada cópia ou parte destes programas de exemplo ou qualquer trabalho derivado deve incluir um aviso de copyright com os dizeres:

© (nome da empresa) (ano). Partes deste código são derivadas dos Programas de Exemplo da IBM Corp. © Copyright IBM Corp. _inserir o ano ou os anos_. Todos os direitos reservados.

Se estas informações estiverem sendo exibidas em cópia eletrônica, as fotografias e ilustrações coloridas podem não aparecer.

Marcas Registradas

IBM, o logotipo IBM, e ibm.com são marcas registradas da International Business Machines Corp., registradas em muitas jurisdições no mundo todo. Outros nomes de produtos e serviços podem ser marcas registradas da IBM ou de outras empresas. Uma lista atual das marcas registradas da IBM está disponível na web emwww.ibm.com/legal/copytrade.shtml.

Os seguintes termos são marcas comerciais e marcas registradas de outras empresas:

Adobe é uma marca registrada da Adobe Systems Incorporated nos Estados Unidos, e/ou outros países.

IT Infrastructure Library é uma marca registrada da Central Computer and Telecommunications Agency que agora faz parte do Office of Government Commerce.

Intel, o logotipo Intel, Intel Inside, o logotipo Intel Inside, Intel Centrino, o logotipo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium, e Pentium são marcas ou marcas registradas da Intel Corporation ou de suas subsidiárias nos Estados Unidos e/ou em outros países.

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

Microsoft, Windows, Windows NT e o logotipo Windows são marcas registradas da Microsoft Corporation nos Estados Unidos e/ou em outros países.

ITIL é uma marca registrada, e uma marca da comunidade registrada do Departamento de Comércio do Governo, e é registrada no U.S. Patent and Trademark Office

UNIX é uma marca registrada da Open Group nos Estados Unidos e/ou em outros países.

Cell Broadband Engine é uma marca registrada da Sony Computer Entertainment, Inc. nos Estados Unidos e/ou em outros países e é utilizada sob licença da mesma.

Java e todas as marcas registradas e logotipos baseados em Java são marcas ou marcas registradas da Oracle e/ou suas afiliadas.

O Serviço Postal dos Estados Unidos possui as seguintes marcas registradas: CASS, CASS Certified, DPV, LACS^{Link}, ZIP, ZIP + 4, ZIP Code, Post Office, Postal Service, USPS e United States Postal Service. IBM Corporation é um licenciado não exclusivo DPV e LACS^{Link} do Serviço Postal dos Estados Unidos.

Outros nomes de empresa, produto ou serviço podem ser marcas registradas ou marcas de serviço de terceiros.

Índice Remissivo

A

Acessando Origens de Dados do Oracle
 acessando as pastas de trabalho do Microsoft Excel 135
 ajustando com o utilitário de ajuste de ODBC 132
 arquivos de biblioteca do wrapper 124
 configurando acesso para origens de dados do IBM InfoSphere Classic Federation Server z/OS 139, 141
 configurando o acesso às origens de dados ODBC 119
 configurando o servidor federado 122
 configurando servidores federados 121
 definindo mapeamentos de usuário 128
 exemplos CREATE NICKNAME 131
 exemplos CREATE SERVER 126
 exemplos CREATE USER MAPPING 128
 exemplos de CREATE WRAPPER 124
 mapeamento de tipos de encaminhamento padrão do Unicode 382
 mapeamento de tipos reversos padrão do Unicode 383
 mapeamentos padrão de tipos de dados de encaminhamento 364
 mapeamentos padrão de tipos de dados reversos 377
 permissões de acesso 138
 registrando 123
 registrando apelidos 131
 registrando definições do servidor 126
 tabelas de teste para o utilitário de ajuste de ODBC 134
 testando o acesso 129
acessibilidade 387
apelidos
 CREATE NICKNAME
 exemplos 189
 criando 4
 Origens de dados de serviços da Web na linha de comandos DB2 230
 Serviços da Web 231
 descrição
 expressões XPath 239
 scripts 170
 opções
 arquivos com estrutura de tabela 333
 banco de dados DB2 281
 BioRS 277
 Script 318
 Serviços da Web 335

apelidos (*continuação*)
 opções (*continuação*)
 XML 342
 origens de dados de serviços da Web
 linha de comando do DB2 230
Origens de dados JDBC 105
Origens de Dados Oracle 159
Origens de Dados Sybase 188
Origens de Dados Teradata
 registrando 215
 servidores federados 216
registrando
 tabelas DB2 70
registrando para as origens de dados ODBC 131
registrando para origens de dados do Microsoft SQL Server 117
registrando para scripts 170
registrando para tabelas, visualizações e sinônimos do Informix 95
técnicas de banco de dados
 Wrapper de Serviços da Web 241
 WSDL (Web Services Description Language) 219
apelidos XML
 visualizações federadas 267
arquivo db2dj.ini 11
 restrições 13
Arquivo de configuração do cliente Informix
 configurando e testando 83
arquivo de configuração do cliente Sybase
 configurando e testando 180, 181
arquivos com estrutura de tabela
 opções de apelido 333
 opções de coluna 333
 opções de wrapper 333
 opções do servidor 333
 recursos federados suportados 273
 Suporte a Unicode 21, 22
 tipos de dados, suportados 384
arquivos de biblioteca
 nomes de arquivo 13
 vinculando 13, 15
arquivos de biblioteca do wrapper
 nomes de arquivo 13
arquivos de configuração
 cliente Informix
 configurando e testando 83
arquivos Excel
 preparando para acessar 72
 tipos de dados, suportados 384
aspas
 uso em instruções SQL 25
avisos legais 389

B

banco de dados
 descrição 29

banco de dados
 catalogando
 remotamente 62
banco de dados federados
 conjuntos de códigos 17
 criação 17
 seqüência de intercalação 17
bibliotecas
 Wrapper BioRS 33
 wrapper de script 168
 wrapper do arquivo estruturado em tabela 194
 Wrapper do Excel 76
 wrapper do XML 251
 wrapper Teradata 209
bibliotecas de wrapper 227
 nomes de arquivo Teradata 209
 nomes de arquivos BioRS 33
 nomes de arquivos do Excel 76
 nomes de arquivos estruturados em tabela 194
 nomes do arquivo de script 168
 XML 251
Biomax Informatics 29

C

cabeçalhos SOAP
 acesso federado 237
caracteres maiúsculos
 uso em instruções SQL 25
caracteres minúsculos
 uso em instruções SQL 25
catalog
 catálogo global 349
catálogo global
 visualizações contendo informações federadas 349
cenários
 acessando arquivos do Excel 80
chaves de licença
 Federação 16
CLP (Processador de Linha de Comandos)
 iniciando 10
comando db2fedsrvfcg
 criando tabelas de teste 134
 exemplos 133
 sintaxe 133
comando db2script_daemon 166
comandos
 emitindo 10
conectividade
 resolução de problemas para origens de dados Oracle 157
 testando conexões do servidor Oracle 157
conexões
 resolução de problemas 69, 93, 104, 116, 130, 158, 187, 214
 testando 68

- configuração
 - ferramenta 1
 - servidores federados 28
- conjuntos de caracteres
 - resolução de problemas de
 - configurações da origem de dados
 - Teradata 208, 218
- conjuntos de códigos
 - banco de dados federado 17
- consultas
 - acessando origens de dados
 - Serviços da Web 244
 - acessando origens de dados
 - BioRS 45, 51
 - acessando origens de dados do
 - Excel 79
 - consultas XML 270
 - exemplos
 - wrappers de serviços da Web 241
 - operações de igualdade do BioRS 48
 - origens de dados de script, acessando
 - origens de dados 173
 - consultas de amostra
 - origens de dados BioRS 51
 - contextos confiáveis federados
 - origens de dados suportadas 273
 - CREATE SERVER
 - Origens de dados JDBC 101
 - CREATE USER MAPPING
 - Origens de dados JDBC 103
 - criação de logs de erro
 - desativando a criação de logs de
 - acesso do Teradata 219

D

- DAEMON_LOGFILE_DIR
 - daemon de script 163
- DAEMON_PORT
 - daemon de script 163
- daemons
 - arquivo de configuração
 - exemplos de script 163
 - configurando para script 163
 - iniciando por script 165
- data
 - importando 273
- Data Studio 2, 3
 - Administration Explorer 3, 4, 7
 - Editor de SQL e XQuery 8
- DB2 databases
 - opções de apelido 281
 - opções de coluna 281
 - opções de mapeamento de
 - usuário 281
 - opções de wrapper 281
 - opções do servidor 281
 - recursos federados suportados 273
- DB2 for Linux, UNIX, and Windows
 - mapeamentos padrão de tipos de
 - encaminhamento 357
 - mapeamentos padrão de tipos
 - reversos 371
- DB2 para System i
 - mapeamentos padrão de tipos de
 - encaminhamento 357

- DB2 para System i (*continuação*)
 - mapeamentos padrão de tipos
 - reversos 371
- DB2 para VM e VSE
 - mapeamentos padrão de tipos de
 - encaminhamento 357
 - mapeamentos padrão de tipos
 - reversos 371
- DB2 para z/OS e OS/390
 - mapeamentos padrão de tipos de
 - encaminhamento 357
 - mapeamentos padrão de tipos
 - reversos 371
- definições de servidor
 - criando 4
 - Origens de dados JDBC 100
 - registrando 64
 - registrando origens de dados do
 - Informix
 - linha de comando do DB2 89
 - registrando para as origens de dados
 - ODBC 126
 - registrando para origens de dados do
 - Microsoft SQL Server 112
 - registrando para origens de dados
 - OLE DB 144
 - registrando para origens de dados
 - Oracle 153
 - registrando para origens de dados
 - Sybase 183
 - registrando para origens de dados
 - Teradata 210
 - Serviços da Web 227
 - distinção entre maiúsculas e minúsculas
 - preservando 25
 - documentação
 - acessível 387
 - driver ODBC do Excel
 - acessando as pastas de trabalho do
 - Microsoft Excel 135
 - permissões de acesso 138
- E
 - elemento AllText
 - BioRS 51
 - erros de operação UPDATE ou DELETE
 - resolução de problemas para as
 - origens de dados Teradata 218
 - especificando
 - valores
 - opções 25
 - esquemas XML
 - mapeamento para apelidos 239
 - estatísticas
 - para origens de dados BioRS 57
 - Excel
 - opções de apelido 289
 - opções de wrapper 289
 - opções do servidor 289
 - recursos federados suportados 273
 - exemplos
 - arquivos Excel 71, 72
 - comando db2script_daemon 166
 - consultas
 - Serviços da Web 244
 - consultas BioRS 49, 51

- exemplos (*continuação*)
 - consultas do Excel 79
 - CREATE SERVER
 - Informix 90
 - Teradata 211
 - exemplos XML 270
 - instrução CREATE NICKNAME,
 - arquivo estruturado em tabela 197
 - instrução CREATE NICKNAME,
 - BioRS 42
 - instrução CREATE NICKNAME,
 - Excel 78
 - instrução CREATE SERVER, arquivo
 - estruturado em tabela 195
 - instrução CREATE SERVER,
 - BioRS 35
 - instrução CREATE SERVER, Excel 77
 - instrução CREATE SERVER, serviços
 - da Web 228
 - instrução CREATE SERVER,
 - XML 254
 - Instrução CREATE USER MAPPING
 - Informix 92
 - Origens de Dados Sybase 186
 - instrução CREATE USER MAPPING,
 - BioRS 39
 - instrução CREATE VIEW, XML 267
 - instrução CREATE WRAPPER,
 - arquivo estruturado em tabela 194
 - instrução CREATE WRAPPER,
 - BioRS 34
 - instrução CREATE WRAPPER,
 - Excel 76
 - instrução CREATE WRAPPER,
 - XML 252
 - instrução UPDATE, BioRS 59
 - mapeamento de documentos
 - XML 247
 - origens de dados BioRS
 - servidor proxy, exemplo do
 - mapeamento de usuário 39
 - origens de dados de script, CREATE
 - NICKNAME 170
 - origens de dados de script, CREATE
 - SERVER 169
 - servidor proxy
 - exemplo de mapeamento de
 - usuário BioRS 39
 - wrappers de serviços da Web 231,
 - 241
 - exemplos CREATE NICKNAME
 - Acessando Origens de Dados do
 - Oracle 131
 - Origens de Dados Microsoft SQL
 - Server 118
 - Origens de Dados Oracle 159
 - exemplos CREATE SERVER
 - Origens de Dados Microsoft SQL
 - Server 113
 - Origens de Dados OLE DB 144
 - Origens de Dados Oracle 154
 - Origens de Dados Sybase 184
 - exemplos CREATE USER MAPPING
 - Acessando Origens de Dados do
 - Oracle 128
 - origens de dados do Oracle 156

exemplos CREATE USER MAPPING
(*continuação*)
 Origens de Dados Microsoft SQL
 Server 115
 Origens de Dados OLE DB 146
 Origens de Dados Teradata 213

F

Federação
 ativando 11
 instalação 1
 resolução de problemas 15
ferramenta de configuração 1
função WSSCRIPT.ARGS 162
funções customizadas
 BioRS 45
 BioRS, descrições 31
 BioRS, registrando 31
 script 162

I

IDs de usuário
 distinção entre maiúsculas e
 minúsculas 25
indicadores de funcionamento federados
 origens de dados suportadas 273
Informix
 arquivos de biblioteca 15
 Caminho padrão para o diretório 88
 conversão da página de códigos 86
 Exemplo de Instrução CREATE
 NICKNAME 95
 instrução CREATE SERVER 90
 Instrução CREATE USER
 MAPPING 92
 mapeamentos padrão de tipos de
 encaminhamento 357
 mapeamentos padrão de tipos
 reversos 371
 nomes de arquivos da biblioteca 88
 opções de coluna 290
 opções de mapeamento de
 usuário 290
 opções de wrapper 290
 opções do servidor 290
 recursos federados suportados 273
Informix servers
 testando a conexão 93
instalação de atualização das estatísticas
de apelidos
 origens de dados suportadas 273
instrução CREATE FUNCTION
 sintaxe do BioRS 31
Instrução CREATE NICKNAME
 exemplos, Excel 78
 exemplos de arquivos estruturados em
 tabela 197
 exemplos do BioRS 42
 exemplos XML 262
 Informix 95
 origens de dados de script 170
 origens de dados de serviços da
 Web 230
 Origens de dados JDBC 106

Instrução CREATE NICKNAME
(*continuação*)
 resolução de problemas para as
 origens de dados Teradata 217
 sintaxe, Excel 77
 sintaxe do arquivo estruturado em
 tabela 196
 sintaxe do BioRS 41
 Teradata 216

instrução CREATE SERVER
 exemplos de arquivos estruturados em
 tabela 195
 exemplos para ODBC 126
 exemplos XML 254
 Informix 90
 sintaxe, Excel 77
 sintaxe do arquivo estruturado em
 tabela 195
 sintaxe do BioRS 35
 sintaxe XML 253
 Teradata 211

Instrução CREATE SERVER
 exemplo, Excel 77
 exemplo de serviços da Web 228
 exemplos do BioRS 35
 origens de dados da família DB2 65
 origens de dados de script 169
 Origens de Dados Sybase 184

Instrução CREATE USER MAPPING
 exemplos do BioRS 39
 Informix 92
 Origens de Dados Sybase 186
 sintaxe do BioRS 38
instrução CREATE VIEW
 exemplos XML 267
instrução CREATE WRAPPER
 Excel, exemplo 76
 exemplo de arquivo estruturado em
 tabela 194
 exemplos do BioRS 34
 exemplos para ODBC 124
 exemplos XML 252
 Origens de dados JDBC 100
 registrando origens de dados OLE
 DB 143

 sintaxe do arquivo estruturado em
 tabela 194
 sintaxe do BioRS 32
 sintaxe XML 250
instrução UPDATE
 exemplo, cardinalidade de apelido
 BioRS 58
 exemplo, cardinalidade de coluna
 ID do BioRS 60
 exemplo, cardinalidade de coluna
 BioRS 59
 exemplos
 instrução UPDATE, BioRS 58

intercalando seqüências
 banco de dados federado 17
isolamento de nível de conexão
 origens de dados suportadas 273
isolamento de nível de instrução
 origens de dados suportadas 273

J

JDBC
 opções de coluna 295
 opções de mapeamento de
 usuário 295
 opções de wrapper 295
 opções do servidor 295
 recursos federados suportados 273

L

label-based access control (LBAC)
 origens de dados suportadas 273
leitores de tela 387
licenças
 Federação 16
 políticas
 Federação 16
locais
 default 11

M

mapeamento
 documentos XML 239
mapeamentos de funções
 opções 353
mapeamentos de tipos de dados
 encaminhamento 357
 reverso 371
mapeamentos de tipos de
 encaminhamento
 mapeamentos padrão 357
 Unicode
 Acessando Origens de Dados do
 Oracle 382
 Microsoft SQL Server 381
 origens de dados do NET8 381
 Origens de dados JDBC 380
 Origens de Dados Sybase 383
mapeamentos de tipos reversos
 mapeamentos padrão 371
 Unicode
 Acessando Origens de Dados do
 Oracle 383
 origens de dados do NET8 382
 Origens de dados JDBC 380
 Origens de Dados Microsoft SQL
 Server 381
 Origens de Dados Sybase 384
mapeamentos do usuário
 criando 4, 67
 criando para origens de dados do
 Informix 92
 criando para origens de dados do
 Sybase 186
definindo as origens de dados
 Oracle 155
definindo para as origens de dados
 ODBC 128
definindo para origens de dados do
 Microsoft SQL Server 115
definindo para origens de dados
 Teradata 212
Origens de dados JDBC 103
Origens de Dados OLE DB 145

- mapeamentos padrão de tipos de dados de encaminhamento
 - Acessando Origens de Dados do Oracle 364
 - DB2 Database Linux, UNIX, and Windows data sources 357
 - DB2 System i data sources 358
 - DB2 z/OS data sources 360
 - exemplo do Informix 369
 - exemplo do Microsoft SQL Server 369
 - exemplo do Oracle 370
 - exemplo do Sybase 370
 - exemplo do Teradata 370
 - exemplos 369
 - Origens de dados do DB2 para VM e VSE 359
 - Origens de dados do Microsoft SQL Server 363
 - Origens de dados do Oracle NET8 366
 - Origens de Dados Informix 360
 - Origens de dados JDBC 362
 - Origens de Dados Sybase 366
 - Origens de Dados Teradata 368, 379
- mapeamentos padrão de tipos de dados reversos
 - Acessando Origens de Dados do Oracle 377
 - DB2 Database Linux, UNIX, and Windows data sources 372
 - DB2 System i data sources 373
 - DB2 z/OS data sources 374
 - Origens de dados do DB2 para VM e VSE 373
 - Origens de dados do Microsoft SQL Server 376
 - Origens de dados do Oracle NET8 377
 - Origens de Dados Informix 374
 - Origens de dados JDBC 375
 - Origens de Dados Sybase 378
- marcas registradas
 - lista de 389
- Microsoft Excel
 - workbooks
 - acessando 135
 - permissões de acesso 138
- Microsoft SQL Server
 - arquivos de biblioteca 15
 - mapeamento de tipos de encaminhamento padrão do Unicode 381
 - mapeamentos padrão de tipos de encaminhamento 357
 - mapeamentos padrão de tipos reversos 371
 - opções de coluna 302
 - opções de mapeamento de usuário 302
 - opções de wrapper 302
 - opções do servidor 302
 - recursos federados suportados 273
 - Suporte a Unicode 21

N

- nós
 - catalogando 62
 - DB2 62

O

- Objetos federados
 - Data Studio 2, 3
- ODBC
 - mapeamentos padrão de tipos de encaminhamento 357
 - opções de coluna 307
 - opções de mapeamento de usuário 307
 - opções de wrapper 307
 - opções do servidor 307
 - rastreios 118
 - recursos federados suportados 273
 - Suporte a Unicode 21
 - variáveis de ambiente 11
- opção CODEPAGE 21
- opção de mapeamento de função DISABLE
 - configurações válidas 353
- opção de mapeamento de função REMOTE_NAME
 - configurações válidas 353
- opção do servidor FOLD_ID
 - distinção entre maiúsculas e minúsculas 25
- opção do servidor FOLD_PW
 - distinção entre maiúsculas e minúsculas 25
- opções de coluna
 - arquivos com estrutura de tabela 333
 - banco de dados DB2 281
 - BioRS 277
 - Informix 290
 - JDBC 295
 - Microsoft SQL Server 302
 - ODBC 307
 - Oracle 313
 - Script 318
 - Serviços da Web 335
 - Sybase 323
 - Teradata 329
 - XML 342
- opções de coluna e apelido
 - Wrapper de Serviços da Web 219, 239
- opções de mapeamento de usuário
 - banco de dados DB2 281
 - BioRS 277
 - Informix 290
 - JDBC 295
 - Microsoft SQL Server 302
 - ODBC 307
 - Oracle 313
 - Script 318
 - Serviços da Web 335
 - Sybase 323
 - Teradata 329
 - XML 342
- opções de wrapper
 - arquivos com estrutura de tabela 333

- opções de wrapper (*continuação*)
 - banco de dados DB2 281
 - BioRS 277
 - Excel 289
 - Informix 290
 - JDBC 295
 - Microsoft SQL Server 302
 - ODBC 307
 - Oracle 313
 - Script 318
 - Serviços da Web 335
 - Sybase 323
 - Teradata 329
 - XML 342
- opções do servidor
 - arquivos com estrutura de tabela 333
 - banco de dados DB2 281
 - BioRS 277
 - definindo as opções do servidor FOLD_ID e FOLD_PW 94
 - Excel 289
 - exemplos para OLE DB 144
 - Informix 290
 - JDBC 295
 - Microsoft SQL Server 302
 - ODBC 307
 - Oracle 313
 - Script 318
 - Serviços da Web 335
 - Sybase 323
 - Teradata 329
 - XML 342
- Oracle
 - arquivo de configuração do cliente 151
 - arquivos de biblioteca 15
 - mapeamentos padrão de tipos de encaminhamento 357
 - mapeamentos padrão de tipos reversos 371
 - opções de coluna 313
 - opções de mapeamento de usuário 313
 - opções de wrapper 313
 - opções do servidor 313
 - recursos federados suportados 273
 - testando a conexão do servidor 157
- origens de dados
 - arquivos de mensagem 13
 - configurando 1, 62
 - configurando o acesso ao Informix 82
 - opções 277
 - recursos suportados 273
 - registrando 63
 - registrando definições de servidor Informix
 - linha de comando do DB2 89
 - registrando wrappers do Informix 88
 - resolução de problemas de conexões 69, 93, 104, 116, 130, 158, 187, 214
 - testando a conexão com o servidor Informix 93
 - tipos de servidores válidos 355
 - visão geral de configuração 22

- origens de dados BioRS
 - Caminho padrão para o diretório 33
 - configurando o acesso 29
 - consultas, operações de igualdade 48
 - consultas, utilizando predicados de junção 49
 - consultas de amostra 49
 - curingas, em consultas 45, 51
 - definição do servidor, registrando 35
 - descrição 29
 - desempenho de consultas, otimizando 57
 - desempenho do wrapper, otimizando 56
 - elemento AllText 51
 - estatísticas
 - cardinalidade de apelido, atualizando 58
 - cardinalidade de coluna, atualizando 59
 - cardinalidade de coluna `_ID_`, atualizando 60
 - estatísticas de cardinalidade, apelidos 58
 - estatísticas de cardinalidade, determinando 58
 - Exemplos da instrução CREATE SERVER 35
 - exemplos da instrução CREATE WRAPPER 34
 - Exemplos de instrução CREATE NICKNAME 42
 - federated features 273
 - funções customizadas
 - descrição 31
 - registrando 31
 - visão geral 45
 - incluindo BioRS em um sistema federado 30
 - informações estatísticas 57
 - mapeamentos de usuário, criando 38
 - nomes de bibliotecas do wrapper 33
 - opções de apelido 277
 - opções de coluna 277
 - opções de mapeamento de usuário 277
 - opções de wrapper 277
 - opções do servidor 277
 - otimizando o desempenho do wrapper 57
 - pré-requisitos 30
 - programa utilitário `admin_find` 58
 - programa utilitário `www_find.cgi` 58
 - registrando apelidos 41
 - servidor proxy
 - exemplo de definição de servidor 35
 - exemplo de definição do servidor BioRS 35
 - exemplo de wrapper 34
 - sintaxe da instrução CREATE FUNCTION 31
 - sintaxe da instrução CREATE NICKNAME 41
 - sintaxe da instrução CREATE SERVER 35
- origens de dados BioRS (*continuação*)
 - Sintaxe da instrução CREATE USER MAPPING 38
 - sintaxe da instrução CREATE WRAPPER 32
 - tipos de dados suportados 384
 - wrapper, registrando 32
- origens de dados de arquivos estruturados em tabela 191
 - apelidos, registrando 196
 - atributos de tipo de arquivo 192
 - autorizações para acesso a arquivos 201
 - Caminho padrão para o diretório 194
 - definição do servidor, registrando 195
 - exemplo de instrução CREATE WRAPPER 194
 - Exemplos da instrução CREATE SERVER 195
 - Exemplos de instrução CREATE NICKNAME 197
 - incluindo em um sistema federado 194
 - nomes de bibliotecas do wrapper 194
 - otimização de consulta 202
 - pré-requisitos 194
 - sintaxe da instrução CREATE NICKNAME 196
 - sintaxe da instrução CREATE SERVER 195
 - sintaxe da instrução CREATE WRAPPER 194
 - visão geral do NIS 191
 - visão geral do wrapper 192
 - wrapper, registrando 194
- origens de dados de script
 - consultas de amostra 173
 - definição do servidor 169
 - Exemplo de Instrução CREATE NICKNAME 170
 - Instrução CREATE SERVER 169
 - nomes de bibliotecas do wrapper 168
 - opções de apelido 171
 - otimizando o desempenho 175
 - registrando apelidos 170
- origens de dados de serviços da Web
 - registrando apelidos (linha de comandos do DB2) 230
- Origens de Dados do Banco de Dados DB2 para Linux, UNIX e Windows
 - mapeamentos padrão de tipos de dados de encaminhamento 357
 - mapeamentos padrão de tipos de dados reversos 372
- origens de dados do DB2
 - apelidos 70
 - arquivos de biblioteca 64
 - Caminho padrão para o diretório 64
 - configurando o acesso 4
 - criando apelidos 70
 - Instrução CREATE SERVER 65
 - mapeamentos do usuário 67
 - registrando 63
 - registrando servidores 64
 - testando conexões 68
- Origens de dados do DB2 para System i
 - mapeamentos padrão de tipos de dados de encaminhamento 358
 - mapeamentos padrão de tipos de dados reversos 373
- Origens de dados do DB2 para VM e VSE
 - mapeamentos padrão de tipos de dados de encaminhamento 359
 - mapeamentos padrão de tipos de dados reversos 373
- Origens de dados do DB2 para z/OS
 - mapeamentos padrão de tipos de dados de encaminhamento 360
 - mapeamentos padrão de tipos de dados reversos 374
- origens de dados do Microsoft SQL
 - definindo mapeamentos de usuário 115
- origens de dados do NET8
 - mapeamento de tipos reversos padrão do Unicode 382
 - mapeamentos de tipos de encaminhamento padrão do Unicode 381
- origens de dados do Oracle
 - arquivos de biblioteca 13
 - configurando o acesso 4
 - resolução de problemas de conectividade 157
- Origens de dados do Oracle NET8
 - mapeamentos padrão de tipos de dados de encaminhamento 366
 - mapeamentos padrão de tipos de dados reversos 377
- origens de dados do servidor Microsoft SQL
 - acessando 107
- Origens de Dados Excel 71
 - amostra de cenário do usuário 80
 - apelidos, registrando 77
 - autorizações para acesso a arquivos 82
 - Caminho padrão para o diretório 76
 - consultas de amostra 79
 - definição do servidor, registrando 77
 - descrição 71
 - incluindo o Excel em um sistema federado 75
 - instrução CREATE NICKNAME, exemplos 78
 - instrução CREATE SERVER, exemplos 77
 - instrução CREATE WRAPPER, exemplos 76
 - nomes de bibliotecas do wrapper 76
 - pré-requisitos 75
 - wrapper, registrando 76
- Origens de Dados Informix
 - arquivos de biblioteca 13
 - mapeamentos padrão de tipos de dados de encaminhamento 360
 - mapeamentos padrão de tipos de dados reversos 374
 - variáveis de ambiente 11
- Origens de dados JDBC
 - apelidos 105

- Origens de dados JDBC *(continuação)*
 - arquivos de biblioteca do wrapper 99
 - configurando o acesso 96
 - configurando servidores federados 97
 - definindo mapeamentos de usuário 103
 - exemplos CREATE NICKNAME 106
 - exemplos CREATE SERVER 101
 - exemplos CREATE USER MAPPING 103
 - exemplos de CREATE WRAPPER 100
 - mapeamento de tipos de encaminhamento padrão do Unicode 380
 - mapeamento de tipos reversos padrão do Unicode 380
 - mapeamentos padrão de tipos de dados de encaminhamento 362
 - mapeamentos padrão de tipos de dados reversos 375
 - registrando definições do servidor 98, 100
 - testando o acesso 104
 - variáveis de ambiente 11
- Origens de Dados Microsoft SQL Server
 - apelidos 117
 - arquivos de biblioteca 13
 - arquivos de biblioteca do wrapper 111
 - configurando as variáveis de ambiente 109
 - configurando o acesso 106
 - configurando servidores federados 107
 - exemplo CREATE NICKNAME 118
 - exemplos CREATE SERVER 113
 - exemplos CREATE USER MAPPING 115
 - mapeamento de tipos reversos padrão do Unicode 381
 - mapeamentos padrão de tipos de dados de encaminhamento 363
 - mapeamentos padrão de tipos de dados reversos 376
 - registrando 111
 - registrando definições do servidor 112
 - resolvendo problemas de configuração 118
 - testando conexões com o Microsoft SQL Server 116
 - variáveis de ambiente 11, 110
- origens de dados não relacionais
 - tipos de dados suportados 384
- Origens de Dados OLE DB
 - configurando o acesso 142
 - definindo mapeamentos de usuário 145
 - exemplos CREATE SERVER 144
 - exemplos CREATE USER MAPPING 146
 - nomes de arquivos da biblioteca 144
 - opções do servidor 144
 - registrando 143
- Origens de Dados OLE DB *(continuação)*
 - registrando definições do servidor 144
- Origens de Dados Oracle
 - acessando 147
 - arquivos de biblioteca do wrapper 153
 - configurando a conectividade 151
 - configurando variáveis de ambiente 147
 - conversão da página de códigos 150
 - definindo mapeamentos de usuário 155
 - exemplos CREATE NICKNAME 159
 - exemplos CREATE SERVER 154
 - exemplos CREATE USER MAPPING 156
 - incluindo em sistemas federados 147
 - registrando 152
 - registrando apelidos 159
 - registrando definições do servidor 153
 - variáveis de ambiente 11, 148
- Origens de Dados Sybase
 - arquivos de biblioteca 13
 - caminho padrão para o diretório do wrapper 183
 - configurando as variáveis de ambiente 177
 - configurando o acesso 175
 - configurando o arquivo de configuração do cliente 180, 181
 - criando mapeamentos de usuário 186
 - exemplos CREATE NICKNAME 189
 - exemplos CREATE SERVER 184
 - exemplos CREATE USER MAPPING 186
 - Instrução CREATE USER MAPPING 186
 - mapeamento de tipos de encaminhamento padrão do Unicode 383
 - mapeamento de tipos reversos padrão do Unicode 384
 - mapeamentos padrão de tipos de dados de encaminhamento 366
 - mapeamentos padrão de tipos de dados reversos 378
 - nomes dos arquivos de biblioteca de wrapper 183
 - registrando 182
 - registrando apelidos 188
 - registrando definições do servidor 183
 - resolução de problemas das configurações de wrapper 189, 190
 - testando conexões com o servidor Sybase 187
 - variáveis de ambiente 11, 179
- Origens de Dados Teradata
 - ajuste 219
 - apelidos 216
 - arquivos de biblioteca 13
 - configurando o acesso 202
 - configurando variáveis de ambiente 204
- Origens de Dados Teradata *(continuação)*
 - definindo mapeamentos de usuário 212
 - desativando a criação de logs de acesso 219
 - exemplos CREATE USER MAPPING 213
 - mapeamentos padrão de tipos de dados de encaminhamento 368, 379
 - registrando 209
 - registrando apelidos 215
 - registrando definições do servidor 210
 - resolução de problemas da instrução CREATE NICKNAME 217
 - resolução de problemas de apelidos 218
 - resolução de problemas de conectividade 214
 - resolução de problemas de conjuntos de caracteres 208, 218
 - testando a conexão 203
 - testando a conexão com o servidor federado 214
 - testando a configuração do cliente 203
 - variáveis de ambiente 11, 205
 - verificando link de tempo de execução 203
- origens de dados XML 247
 - apelidos 259
 - apelidos, associações de dados 257
 - apelidos, registrando 262
 - Caminho padrão para o diretório 251
 - consultas, exemplos 270
 - definição do servidor, registrando 253
 - espaços de nomes 260
 - Exemplos da instrução CREATE SERVER 254
 - exemplos da instrução CREATE VIEW 267
 - Exemplos de instrução CREATE NICKNAME 262
 - expressões XPath 257
 - incluindo em um sistema federado 250
 - instrução CREATE WRAPPER 252
 - nomes de bibliotecas do wrapper 251
 - opções de apelido
 - analisando documentos XML 257
 - espaços de nomes 260
 - opções de apelidos, otimização de consulta 259
 - otimização de consulta 259
 - otimização de consulta, recurso de modelo de custo 269
 - otimização de consulta, visão geral 266
 - otimização de consulta, visualizações federadas 267
 - pré-requisitos 250
 - recurso de modelo de custo 259
 - servidor de proxy, exemplos de definições de servidores 254
 - servidor proxy, exemplos do wrapper 252

- origens de dados XML (*continuação*)
 - servidores proxy 256
 - tuplas 257
 - visão geral, apelidos 257
 - visão geral do NIS 247
 - wrapper, registrando 250
- otimização
 - origens de dados de script 175
- otimização de assincronia
 - origens de dados suportadas 273
- otimização de consulta
 - orientações para arquivos estruturados em tabela 202
 - orientações para consultas BioRS 57
 - XML 269

P

- páginas de código
 - banco de dados federado 17
 - conversão
 - Informix 86
 - Origens de Dados Oracle 150
 - erros quando diferente 22
 - especificando para suporte Unicode
 - exemplo 22
 - visão geral 21
 - sistemas federados 19
 - suporte para MSSQL, ODBCm e CODEPAGE 21
- parâmetro CONCENTRATOR 11
- parâmetro FEDERATED 11
- pass-through
 - sessions
 - origens de dados suportadas 273
- planejamento
 - acessando dados do Excel 72
 - pontos de salvamento do aplicativo
 - origens de dados suportadas 273
- problemas de conexão
 - Origens de Dados Microsoft SQL Server 118
- procedimentos federados
 - origens de dados suportadas 273
- proxy HTTP
 - origens de dados suportadas 273

R

- repositório de mapeamento de usuário
 - origens de dados suportadas 273
- resolução de problemas
 - acesso à origem de dados 11
 - conexões 69, 93, 104, 116, 130, 158, 187, 214
 - configuração do servidor federado 10, 11
 - Instalação da federação 13, 15
 - servidor federado 13
- restrições
 - arquivo db2dj.ini 13

S

- Script
 - opções de apelido 318

- Script (*continuação*)
 - opções de coluna 318
 - opções de mapeamento de usuário 318
 - opções de wrapper 318
 - opções do servidor 318
 - recursos federados suportados 273
- scripts
 - configurar como origens de dados 160
- segurança
 - Serviços da Web 229
- senhas
 - distinção entre maiúsculas e minúsculas 25
- Serviços da Web
 - apelidos 219, 231
 - arquivos de biblioteca do wrapper 227
 - consultas de amostra 244
 - definições de servidor 227
 - Exemplo de instrução CREATE SERVER 228
 - exemplos 241
 - opções de apelido 335
 - opções de coluna 335
 - opções de mapeamento de usuário 335
 - opções de wrapper 335
 - opções do servidor 335
 - recursos federados suportados 273
 - segurança 229
 - servidor proxy, exemplo de definição do servidor 228
 - tipos de dados, suportados 384
 - visualizações federadas 240
- servidores
 - testando conexões com o Microsoft SQL Server 116
 - testando conexões com o servidor Sybase 187
- servidores federados
 - acessando origens de dados do Microsoft SQL Server 107
 - bancos de dados 17
 - catalogando bancos de dados 62
 - comandos 10
 - configuração 10
 - configurando 1
 - configurando acesso ao Microsoft SQL Server 106
 - configurando acesso às origens de dados Sybase 175
 - configurando acesso para as origens de dados do Microsoft SQL Server 107
 - configurando diversos 7
 - configurando o acesso ao informix 82
 - configurando origens de dados 62
 - configurando para JDBC 97
 - configurando para ODBC 121, 122
 - configurando vários 28
 - resolução de problemas 10
 - testando a conexão com o servidor Teradata 203

- servidores federados (*continuação*)
 - testando o acesso aos servidores da origem de dados ODBC 129
 - testando o acesso aos servidores da origem de dados JDBC 104
 - visão geral de configuração 22
- servidores proxy
 - acesso às origens de dados XML 256
 - exemplo de definição de servidor de serviços da Web 228
 - exemplo de wrapper BioRS 34
- sinônimos do Informix
 - registrando apelidos 95
- SOCKS
 - proxy
 - origens de dados suportadas 273
- SQL10013N
 - resolução de problemas 189
- SQL1097N
 - resolução de problemas 190
- SQL1822N
 - resolução de problemas 190
- SSL
 - origens de dados suportadas 273
- suporte ao idioma nacional (NLS)
 - banco de dados federado 17
- Sybase
 - arquivos de biblioteca 15
 - mapeamentos padrão de tipos de encaminhamento 357
 - mapeamentos padrão de tipos reversos 371
 - opções de coluna 323
 - opções de mapeamento de usuário 323
 - opções de wrapper 323
 - opções do servidor 323
 - recursos federados suportados 273
 - script djlLinkSybase 15

T

- tabelas de cache
 - origens de dados suportadas 273
- tabelas de consulta materializada (MQTs)
 - origens de dados suportadas 273
- tabelas do Informix
 - registrando apelidos 95
- Teradata
 - arquivos de biblioteca 15
 - Exemplo de Instrução CREATE NICKNAME 216
 - instrução CREATE SERVER 211
 - mapeamentos padrão de tipos de encaminhamento 357
 - mapeamentos padrão de tipos reversos 371
 - nomes de bibliotecas do wrapper 209
 - opções de coluna 329
 - opções de mapeamento de usuário 329
 - opções de wrapper 329
 - opções do servidor 329
 - recursos federados suportados 273
 - script djlLinkTeradata 15

- tipos de dados LOB
 - origens de dados suportadas, leitura e gravação 273
 - origens de dados suportadas, somente leitura 273
- tipos de dados
 - para origens de dados não relacionais 384
- tipos de servidores
 - tipos federados válidos 355
- tolerância de erros
 - origens de dados suportadas 273
- two-phase commit
 - origens de dados suportadas 273

U

- Unicode
 - origens de dados suportadas 273
 - páginas de código 21, 22
 - suporte para sistemas federados 19
 - tamanho do ponto de código 22
 - wrappers MSSQL e ODBC 21
- USER de Registro Especial do DB2
 - exemplos para as origens de dados OLE DB 146
- utilitário de ajuste de ODBC
 - ajustando wrappers ODBC 132
 - criando tabelas de teste 134
 - exemplos do comando exdb2fedsvrcfg 133
 - sintaxe do comando db2fedsvrcfg 133

V

- variáveis de ambiente 11
 - configuração
 - clientes de origem de dados 11
 - Variáveis de Ambiente Teradata 204
 - variáveis Oracle 147
 - configurando variáveis de ambiente do Sybase 177
 - configurando variáveis do Microsoft SQL Server 109
 - Informix 85
 - necessário 11
 - origens de dados 11
 - Origens de Dados Microsoft SQL Server 110
 - Origens de Dados Oracle 148
 - Origens de Dados Sybase 179
 - Origens de Dados Teradata 205
 - resolução de problemas da variável de ambiente Sybase 190
 - restrições 13
 - wrappers do Informix 84
- variável de registro DB2_DJ_INI 11
- vinculando
 - arquivos de biblioteca 13
- visualizações do catálogo SYSSTAT 349
- visualizações do Informix
 - registrando apelidos 95
- visualizações federadas
 - Serviços da Web 240

- visualizações SYSCAT
 - global 349

W

- Web Services
 - acesso ao cabeçalho SOAP 237
- wrapper DB2
 - registrando 63
- wrapper de script
 - criando 168
 - funções customizadas 162
 - visão geral 160
- Wrapper de Serviços da Web
 - origens de dados 219
- wrapper DRDA
 - nomes de arquivos da biblioteca 64
- wrappers
 - Arquivos da Biblioteca do Wrapper JDBC 99
 - Arquivos da Biblioteca do Wrapper OLE DB 144
 - Arquivos de Biblioteca do Wrapper ODBC 124
 - Arquivos de Biblioteca do Wrapper Oracle 153
 - bibliotecas de wrapper do Microsoft SQL Server 111
 - configurando acesso a origens de dados Teradata 202
 - configurando os wrappers JDBC 96
 - configurando os wrappers ODBC 119, 139, 141
 - configurando wrappers OLE DB 142
 - criando 4
 - origens de dados de serviços da Web 219, 226
 - registrando os wrappers ODBC 123
 - registrando wrappers do Microsoft SQL Server 111
 - registrando wrappers do Oracle 152
 - registrando wrappers do Sybase 182
 - registrando wrappers JDBC 98
 - registrando wrappers Teradata 209
 - script 160
 - verificando link de tempo de execução para origens de dados Teradata 203
- wrappers de serviços da Web
 - criando 226
- wrappers do Informix
 - aprimorando o desempenho 94
 - criando mapeamentos de usuário 92
 - registrando 88
 - variáveis de ambiente 84, 85
- WSDL (Web Services Description Language)
 - apelidos 219
 - mapeamento para apelidos 239

X

- XML
 - data type
 - supported 384
 - opções de apelido 342
 - opções de coluna 342

- XML (*continuação*)
 - opções de mapeamento de usuário 342
 - opções de wrapper 342
 - opções do servidor 342
- XPath
 - utilizando com apelidos 239



Impresso no Brasil

SC19-3699-00



Spine information:

IBM InfoSphere Federation Server

Versão 10.1

Guia de Configuração para Origens de Dados Federados

