

IBM DB2 Information Integrator



Guia de Configuração da Origem de Dados

Versão 8

IBM DB2 Information Integrator



Guia de Configuração da Origem de Dados

Versão 8

Antes de utilizar estas informações e o produto suportado por elas, leia as informações gerais na seção "Avisos" na página 461.

Este documento contém informações de propriedade da IBM. Ele é fornecido sob um acordo de licença e é protegido por leis de Copyright. As informações contidas nesta publicação não incluem garantias de produto e nenhuma declaração feita neste manual deve ser interpretada como tal.

Você pode solicitar as publicações IBM on-line ou através de seu representante IBM local:

- Para solicitar publicações on-line, consulte o IBM Publications Center em www.ibm.com/shop/publications/order
- Para localizar o representante IBM local, consulte o IBM Directory of Worldwide Contacts em www.ibm.com/planetwide

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

© Copyright International Business Machines Corporation 1998 - 2003. Todos os direitos reservados.

Índice

| | | | |
|--|-----------|--|-----------|
| Sobre este Manual | xi | Lista de Verificação para Planejamento de | |
| Quem Deve Ler este Manual | xi | Configuração do Sistema Federado | 27 |
| Convenções | xi | Lista de Verificação: Regras de | |
| Como Ler Diagramas de Sintaxe | xi | Nomenclatura de Objetos Federados | 27 |
| | | Lista de Verificação: Preservando Valores | |
| | | com Distinção entre Maiúsculas e | |
| | | Minúsculas | 27 |
| | | Lista de Verificação: Estatísticas de Origem | |
| | | de Dados | 27 |
| | | Lista de Verificação: Mapeamentos de | |
| | | Tipos de Dados | 28 |
| | | Lista de Verificação: Mapeamentos de | |
| | | Usuários | 28 |
| | | Lista de Verificação: Wrappers | 29 |
| Capítulo 1. Visão Geral sobre Como | | | |
| Configurar o Acesso às Origens de Dados | 1 | | |
| Trilha Rápida para Configurar Origens de | | | |
| Dados | 1 | | |
| Origens de Dados Suportadas | 3 | | |
| Criar Pseudônimos para cada Objeto de | | | |
| Origem de Dados | 5 | | |
| Incluindo Opções de Colunas ao Criar um | | | |
| Pseudônimo | 7 | | |
| Criando um Pseudônimo em um | | | |
| Pseudônimo | 7 | | |
| Etapas de Configuração Opcionais | 8 | | |
| Capítulo 2. Visão Geral de um Sistema | | | |
| Federado | 9 | | |
| Wrappers e Módulos de Wrappers | 9 | | |
| Definições do Servidor e Opções do Servidor | 10 | | |
| Intercalando Sequências e Configuração de | | | |
| Origem de Dados | 11 | | |
| Mapeamentos de Usuário | 11 | | |
| Pseudônimos e Objetos da Origem de Dados | 12 | | |
| Objetos Válidos da Origem de Dados | 13 | | |
| Opções da Coluna | 14 | | |
| Mapeamentos de Tipos de Dados | 15 | | |
| Mapeamentos de Função | 16 | | |
| Especificações do Índice | 17 | | |
| Capítulo 3. Planejando a Configuração da | | | |
| Origem de Dados Federada | 19 | | |
| Regras de Nomenclatura de Objetos | | | |
| Federados. | 19 | | |
| Preservando Valores com Distinção entre | | | |
| Maiúsculas e Minúsculas em um Sistema | | | |
| Federado | 20 | | |
| Atualizar Estatísticas de Origem de Dados | 22 | | |
| Planejar Mapeamentos de Tipos de Dados | 22 | | |
| Planejar Mapeamentos de Funções | 24 | | |
| Planejar Mapeamentos do Usuário | 25 | | |
| Escolher o Wrapper Correto. | 25 | | |
| | | Capítulo 4. Configurando o Acesso às | |
| | | Origens de Dados da Família DB2 | 31 |
| | | Adicionando Origens de Dados da Família | |
| | | DB2 a Servidores Federados. | 31 |
| | | Catalogando uma Entrada do Nó no Diretório | |
| | | do Nó Federado | 32 |
| | | Catalogando o Banco de Dados Remoto no | |
| | | Diretório do Banco de Dados do Sistema | |
| | | Federado | 33 |
| | | Registrando o Wrapper DB2 | 34 |
| | | Registrando Definições de Servidor para uma | |
| | | Origem de Dados DB2 | 36 |
| | | Instrução CREATE SERVER - Exemplos para | |
| | | o Wrapper DB2 | 37 |
| | | Criando o Mapeamento de Usuário para uma | |
| | | Origem de Dados DB2 | 38 |
| | | Instrução CREATE USER MAPPING - | |
| | | Exemplos para o Wrapper DB2. | 39 |
| | | Testando a Conexão com o Servidor da | |
| | | Origem de Dados DB2 | 40 |
| | | Registrando Pseudônimos para Tabelas e | |
| | | Exibições do DB2 | 42 |
| | | Instrução CREATE NICKNAME - Exemplos | |
| | | para o Wrapper DB2 | 43 |
| | | Ajustando e Resolvendo Problemas de | |
| | | Configuração para Origens de Dados da | |
| | | Família DB2 | 44 |
| | | Aprimorando o Desempenho, Definindo a | |
| | | Variável DB2_DJ_COMM (UNIX) | 44 |

| | |
|---|-----------|
| Capítulo 5. Configurando o Acesso às Origens de Dados do Informix | 47 |
| Adicionando Origens de Dados Informix a Servidores Federados | 47 |
| Definindo e Testando o Arquivo de Configuração do Cliente Informix | 48 |
| Registrando o Wrapper Informix | 50 |
| Registrando Definições de Servidor para uma Origem de Dados Informix | 51 |
| Instrução CREATE SERVER - Exemplos para o Wrapper Informix | 52 |
| Criando o Mapeamento de Usuário para uma Origem de Dados Informix | 54 |
| Instrução CREATE USER MAPPING - Exemplos para o Wrapper Informix | 55 |
| Testando a Conexão com o Servidor Informix | 56 |
| Registrando Pseudônimos para Tabelas, Exibições e Sinônimos do Informix | 58 |
| Instrução CREATE NICKNAME - Exemplos para o Wrapper Informix. | 59 |
| Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Informix | 60 |
| Aprimorando o Desempenho Definindo as Opções de Servidor FOLD_ID e FOLD_PW | 60 |
| Aprimorando o Desempenho Definindo a Variável DB2_DJ_COMM (UNIX) | 60 |
| Capítulo 6. Configurando o Acesso às Origens de Dados do Oracle | 63 |
| Adicionando Origens de Dados Oracle a Servidores Federados | 63 |
| Configurando e Testando o Arquivo de Configuração do Cliente Oracle | 64 |
| Registrando o Wrapper Oracle | 65 |
| Nomes de Wrappers e de Bibliotecas Oracle | 67 |
| Registrando Definições de Servidor para uma Origem de Dados Oracle. | 68 |
| Instrução CREATE SERVER - Exemplos para o Wrapper Oracle | 69 |
| Criando Mapeamentos de Usuários para uma Origem de Dados Oracle. | 71 |
| Instrução CREATE USER MAPPING - Exemplos para o Wrapper Oracle | 71 |
| Testando a Conexão com o Servidor Oracle | 73 |
| Registrando Pseudônimos para Tabelas e Exibições do Oracle | 74 |
| Instrução CREATE NICKNAME - Exemplos para o Wrapper Oracle | 75 |

| | |
|---|----|
| Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Oracle | 76 |
| Aprimorando o Desempenho Definindo a Variável DB2_DJ_COMM (UNIX) | 76 |
| Problemas de Conectividade | 77 |

| | |
|---|-----------|
| Capítulo 7. Configurando o Acesso às Origens de Dados do Sybase | 79 |
| Adicionando Origens de Dados Sybase a Servidores Federados | 79 |
| Configurando e Testando o Arquivo de Configuração do Cliente Sybase | 80 |
| Registrando o Wrapper Sybase. | 82 |
| Registrando Definições de Servidor para uma Origem de Dados Sybase | 83 |
| Instrução CREATE SERVER - Exemplos para o Wrapper Sybase | 84 |
| Criando um Mapeamento de Usuário para uma Origem de Dados Sybase | 87 |
| Instrução CREATE USER MAPPING - Exemplos para o Wrapper Sybase. | 88 |
| Testando a Conexão com o Servidor Sybase | 89 |
| Registrando Pseudônimos para Tabelas e Exibições do Sybase | 90 |
| Instrução CREATE NICKNAME - Exemplos para o Wrapper Sybase | 91 |
| Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Sybase | 92 |
| Aprimorando o Desempenho Definindo a Variável de Ambiente DB2_DJ_COMM (UNIX). | 92 |
| Utilizando CTLIB em Vez de DBLIB | 93 |
| Resolvendo o Erro sp_helpindex | 93 |

| | |
|--|-----------|
| Capítulo 8. Configurando o Acesso às Origens de Dados do Microsoft SQL Server. | 95 |
| Adicionando Origens de Dados Microsoft SQL Server a Servidores Federados | 95 |
| Preparando o Servidor Federado e o Banco de Dados para Acessarem Origens de Dados Microsoft SQL Server | 96 |
| Registrando o Wrapper Microsoft SQL Server | 98 |
| Registrando Definições de Servidor para uma Origem de Dados Microsoft SQL Server. | 99 |
| Instrução CREATE SERVER - Exemplos para o Wrapper Microsoft SQL Server. | 101 |
| Criando um Mapeamento de Usuário para uma Origem de Dados Microsoft SQL Server | 103 |

| | | | |
|---|------------|--|------------|
| Instrução CREATE USER MAPPING - Exemplos para o Wrapper Microsoft SQL Server. | 104 | Testando a Conexão com o Servidor Teradata Verificando se a Biblioteca Teradata Está Ativada para Link de Tempo de Execução (AIX) | 129 130 |
| Testando a Conexão com o Servidor Remoto Microsoft SQL Server | 105 | Registrando o Wrapper Teradata | 131 |
| Registrando Pseudônimos para Tabelas e Exibições do Microsoft SQL Server | 106 | Registrando Definições de Servidor para uma Origem de Dados Teradata | 132 |
| Instrução CREATE NICKNAME - Exemplos para o Wrapper Microsoft SQL Server | 107 | Instrução CREATE SERVER - Exemplos para Wrapper Teradata | 134 |
| Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Microsoft SQL Server | 108 | Criando o Mapeamento de Usuário para uma Origem de Dados Teradata | 135 |
| Aprimorando o Desempenho Definindo a Variável DB2_DJ_COMM (UNIX) | 108 | Instrução CREATE USER MAPPING - Exemplos para o Wrapper Teradata | 136 |
| Obtendo Rastreios ODBC | 109 | Testando a Conexão do Servidor Federado com o Servidor Teradata | 137 |
| Capítulo 9. Configurando o Acesso às Origens de Dados do ODBC | 111 | Pseudônimos Teradata em Servidores Federados | 138 |
| Adicionando Origens de Dados ODBC a Servidores Federados. | 111 | Registrando Pseudônimos para Tabelas e Exibições do Teradata | 139 |
| Preparando o Servidor Federado e o Banco de Dados para Acessarem Origens de Dados através do ODBC | 113 | Instrução CREATE NICKNAME - Exemplos para Wrapper Teradata | 140 |
| Registrando o Wrapper ODBC | 114 | Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Teradata | 141 |
| Instrução CREATE WRAPPER - Exemplos para o Wrapper ODBC | 115 | Aprimorando o Acesso ao Servidor Teradata Definindo a Variável DB2_DJ_COMM | 141 |
| Registrando Definições de Servidor para uma Origem de Dados ODBC. | 116 | Ajustando e Desativando o Registro de Acesso de Teradata | 142 |
| Instrução CREATE SERVER - Exemplos de Wrapper ODBC | 117 | Ativando o Link de Tempo de Execução para libcliv2.so (AIX) | 142 |
| Criando um Mapeamento de Usuário para uma Origem de Dados ODBC. | 119 | Capítulo 11. Configurando o Acesso às Origens de Dados do OLE DB | 145 |
| Instrução CREATE USER MAPPING - Exemplos para o Wrapper ODBC | 119 | Adicionando Origens de Dados OLE DB a Servidores Federados | 145 |
| Testando a Conexão com o Servidor da Origem de Dados ODBC | 120 | Registrando o Wrapper OLE DB | 146 |
| Registrando Pseudônimos para Tabelas e Exibições de Origens de Dados ODBC | 122 | Registrando Definições de Servidor para uma Origem de Dados OLE DB | 147 |
| Instrução CREATE NICKNAME - Exemplos para o Wrapper ODBC | 123 | Instrução CREATE SERVER - Exemplos para o Wrapper OLE DB | 148 |
| Ajustando e Resolvendo Problemas de Configuração para Origens de Dados ODBC. | 124 | Criando um Mapeamento de Usuário para uma Origem de Dados OLE DB | 149 |
| Aprimorando o Desempenho Definindo a Variável DB2_DJ_COMM | 124 | Instrução CREATE USER MAPPING - Exemplos para o Wrapper OLE DB | 149 |
| Obtendo Rastreios ODBC | 125 | Capítulo 12. Configurando o Acesso às Origens de Dados de Arquivos com Estrutura de Tabela | 151 |
| Capítulo 10. Configurando o Acesso às Origens de Dados do Teradata | 127 | | |
| Adicionando Origens de Dados Teradata a Servidores Federados | 127 | | |

| | | | |
|---|------------|--|------------|
| O que São Arquivos com Estrutura de Tabela? | 151 | Compreendendo Pseudo Colunas | 174 |
| Tipos de Arquivos com Estrutura de Tabela | 152 | Instrução CREATE NICKNAME - Exemplo para o Wrapper Documentum | 179 |
| Arquivos Ordenados. | 152 | Registrando Funções Personalizadas para Origens de Dados do Documentum. | 181 |
| Arquivos Não-ordenados | 152 | Regras de Argumento da Cadeia de Função Personalizada | 182 |
| Como o DB2 Information Integrator Funciona com Arquivos com Estrutura de Tabela. | 152 | Utilizando Funções Personalizadas em Consultas | 182 |
| Incluindo Arquivos com Estrutura de Tabela em um Sistema Federado | 154 | Tabela de Função Personalizada | 183 |
| Registrando o Wrapper do Arquivo Estruturado de Tabela | 155 | Origem de Dados de Documentum – Consultas de Exemplo | 188 |
| Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper de Arquivo Estruturado de Tabela | 156 | O que É o Utilitário CreateNicknameFile para o Wrapper do Documentum? | 189 |
| Registrando o Servidor para Arquivos com Estrutura de Tabela | 156 | Instalando o Utilitário CreateNicknameFile (Wrapper do Documentum) | 190 |
| Registrando Pseudônimos para Arquivos com Estrutura de Tabela | 157 | Configurando o Utilitário CreateNicknameFile (Wrapper do Documentum) | 191 |
| Instrução CREATE NICKNAME - Exemplos para o Wrapper do Arquivo Estruturado Tabela. | 158 | Mapeando o Tipo de Objeto DM_ID em Tabelas Registradas do Documentum | 192 |
| Modelo de Controle de Acesso de Arquivo para o Wrapper do Arquivo Estruturado de Tabela. | 159 | Definição Dupla dos Atributos de Repetição (Wrapper do Documentum) | 193 |
| Dicas e Considerações de Otimização para o Wrapper do Arquivo Estruturado de Tabela . | 160 | Controle de Acesso para o Wrapper do Documentum | 194 |
| Mensagens para o Wrapper do Arquivo Estruturado de Tabela | 160 | Mensagens para o Wrapper do Documentum | 194 |
| Capítulo 13. Configurando o Acesso às Origens de Dados do Documentum | 165 | Capítulo 14. Configurando o Acesso às Origens de Dados do Excel. | 203 |
| O que É Documentum? | 165 | O que É Excel?. | 203 |
| Incluindo o Documentum em um Sistema Federado. | 167 | Incluindo Excel a um Sistema Federado | 205 |
| Disponibilizando a Biblioteca do Cliente Documentum para o DB2 (Apenas AIX e Solaris Operating Environment) | 168 | Registrando o Wrapper do Excel. | 205 |
| Apontando para o Arquivo dmcl.ini do Cliente do Documentum | 169 | Registrando o Servidor para uma Origem de Dados do Excel | 206 |
| Registrando o Wrapper do Documentum | 171 | Registrando Pseudônimos para Origens de Dados do Excel | 207 |
| Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper Documentum | 171 | Origem de Dados Excel – Consultas de Exemplo | 207 |
| Registrando o Servidor para Origens de Dados do Documentum. | 172 | Origem de Dados Excel – Cenário de Amostra | 208 |
| Mapeando Usuários (Wrapper do Documentum) | 173 | Modelo de Controle de Acesso de Arquivo para o Wrapper do Excel | 210 |
| Registrando Pseudônimos para Origens de Dados do Documentum. | 174 | Mensagens para o Wrapper do Excel | 211 |
| | | Capítulo 15. Configurando o Acesso às Origens de Dados do BLAST | 219 |
| | | O que É BLAST? | 219 |
| | | Incluindo BLAST a um Sistema Federado | 224 |

| | |
|---|-----|
| Verificando se a Versão Correta do Executável Blastall e os Arquivos de Matriz Estão Instalados | 225 |
| Configurando o Daemon do BLAST. | 225 |
| Iniciando o Daemon do BLAST | 229 |
| Registrando o Wrapper do BLAST | 230 |
| Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper BLAST | 230 |
| Registrando um Servidor para uma Origem de Dados do BLAST | 231 |
| Registrando Pseudônimos para Origens de Dados do BLAST | 232 |
| Análise de Linha de Definição | 232 |
| Colunas Fixas | 233 |
| Instrução CREATE NICKNAME - Exemplos para Wrapper BLAST | 238 |
| Configurando o TurboBlast para Funcionar com o Wrapper BLAST | 239 |
| Construindo Consultas SQL BLAST. | 240 |
| Origem de Dados BLAST – Consultas de Exemplo | 241 |
| Dicas de Otimização para o Wrapper do BLAST | 243 |
| Mensagens para o Wrapper do BLAST. | 243 |

Capítulo 16. Configurando o Acesso às Origens de Dados do XML 247

| | |
|--|-----|
| O que É XML?. | 247 |
| Incluindo XML a um Sistema Federado | 251 |
| Registrando o Wrapper do XML | 252 |
| Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper XML | 253 |
| Registrando o Servidor para uma Origem de Dados XML. | 254 |
| Associações de Dados entre Pseudônimos e Documentos XML. | 254 |
| O Recurso de Modelo de Custo para o Wrapper XML | 257 |
| Dicas de Otimização para o Recurso de Modelo de Custo XML | 258 |
| Registrando Pseudônimos para Origens de Dados XML. | 260 |
| Instrução CREATE NICKNAME - Exemplos para o Wrapper XML | 261 |
| Criando Exibições Federadas para Pseudônimos Não-raiz (Wrapper XML) | 266 |
| Origem de Dados XML - Consultas de Exemplo | 267 |

| | |
|---|-----|
| Mensagens para o Wrapper do XML | 268 |
|---|-----|

Capítulo 17. Configurando o Acesso às Origens de Dados do Entrez 279

| | |
|---|-----|
| O que É Entrez? | 279 |
| Adicionando Entrez a um Sistema Federado | 280 |
| Registrando Funções Personalizadas para o Wrapper Entrez | 281 |
| Registrando o Wrapper Entrez | 282 |
| Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper Entrez | 283 |
| Registrando o Servidor para uma Origem de Dados Entrez | 283 |
| Registrando Pseudônimos para Origens de Dados Entrez | 284 |
| Funções Personalizadas e Consultas Entrez | 286 |
| Predicados Relacionais para o Wrapper Entrez | 288 |
| Cláusulas WHERE Inválidas para o Wrapper Entrez | 288 |
| Simplificação de Elementos de Dados de Esquema | 289 |
| Listas de Itens | 289 |
| Nomes | 289 |
| Datas | 289 |
| Origem de Dados Entrez — Consultas de Exemplo | 290 |
| Tabela de Funções Personalizadas - Wrapper Entrez | 291 |
| Tabelas de Esquemas PubMed e Nucleotide | 292 |
| Mensagens para o Wrapper Entrez | 300 |

Capítulo 18. Configurando o Acesso às Origens de Dados do Extended Search 307

| | |
|---|-----|
| O que É Extended Search? | 307 |
| Origens de Dados Extended Search | 308 |
| Como Funciona o Wrapper Extended Search | 309 |
| Pseudônimos do Extended Search | 310 |
| Tabelas Verticais do Extended Search | 312 |
| Adicionando Origens de Dados Extended Search a um Servidor Federado | 315 |
| Registrando o Wrapper Extended Search | 316 |
| Registrando o Servidor para Origens de Dados Extended Search | 316 |
| Registrando Pseudônimos para Origens de Dados Extended Search | 317 |
| Registrando Mapeamentos de Usuários para Origens de Dados Extended Search | 318 |

| | |
|--|-----|
| Registrando a Função Personalizada do Extended Search | 320 |
| Wrapper Extended Search - Diretrizes de Consultas | 321 |
| Wrapper Extended Search - Consultas de Exemplo | 323 |
| Wrapper Extended Search - Linguagem de Consulta Generalizada | 326 |
| Mensagens para o Wrapper Extended Search | 328 |

Capítulo 19. Configurando o Acesso às Origens de Dados do HMMER 333

| | |
|---|-----|
| O que É HMMER? | 333 |
| Adicionando HMMER a um Sistema Federado. | 337 |
| Verificando se a Versão Correta do Executável hmpfam Está Instalada | 338 |
| Configurando o Daemon HMMER | 338 |
| Iniciando o Daemon HMMER. | 341 |
| Registrando o Wrapper HMMER. | 342 |
| Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper HMMER. | 343 |
| Registrando o Servidor para uma Origem de Dados HMMER | 344 |
| Argumentos | 344 |
| Opções | 344 |
| Registrando Pseudônimos para Origens de Dados HMMER | 345 |
| Opções de Pseudônimos | 346 |
| Colunas Fixas | 346 |
| Instrução CREATE NICKNAME - Exemplo para o Wrapper HMMER | 349 |
| Origem de Dados HMMER – Exemplo Completo | 349 |
| Construir Novas Consultas HMMER com Amostras | 350 |
| Dicas de Otimização para o Wrapper HMMER. | 351 |
| Mensagens para o Wrapper HMMER | 352 |

| | |
|---|-----|
| Capítulo 20. Alterando pseudônimos 355 | |
| Alterando Pseudônimos. | 355 |
| Alterando o Tipo de Dados | 355 |
| Alterando a Opção de Pseudônimo | 356 |

Capítulo 21. Referência de Comandos DDL 357

| | |
|--|-----|
| Sintaxe da Instrução CREATE FUNCTION - Wrapper Extended Search | 357 |
|--|-----|

| | |
|---|-----|
| Sintaxe da Instrução CREATE NICKNAME - Wrapper BLAST | 358 |
| Sintaxe da Instrução CREATE NICKNAME - Wrapper Documentum | 360 |
| Sintaxe da Instrução CREATE NICKNAME - Wrapper Excel | 364 |
| Sintaxe da Instrução CREATE NICKNAME - Wrapper Extended Search | 366 |
| Opções da Instrução CREATE NICKNAME - Wrapper Entrez | 369 |
| Sintaxe da Instrução CREATE NICKNAME - Wrapper do Arquivo Estruturado de Tabela | 370 |
| Sintaxe da Instrução CREATE NICKNAME - Wrapper XML | 375 |
| Argumentos da Instrução CREATE SERVER - Wrapper BLAST. | 380 |
| Argumentos e Opções da Instrução CREATE SERVER - Wrapper Documentum | 381 |
| Argumentos da Instrução CREATE SERVER - Wrapper Entrez | 382 |
| Argumentos da Instrução CREATE SERVER - Wrapper Excel | 383 |
| Sintaxe da Instrução CREATE SERVER - Wrapper Extended Search | 383 |
| Opções da Instrução CREATE USER MAPPING - Wrapper Documentum | 384 |
| Sintaxe da Instrução CREATE USER MAPPING - Wrapper Extended Search | 385 |
| Sintaxe da Instrução CREATE WRAPPER - Wrapper Extended Search | 386 |

Apêndice A. Exibições da Tabela de Catálogo Global Contendo Informações Federadas 387

Apêndice B. Opções do Servidor para Sistemas Federados 391

Apêndice C. Opções de Mapeamento de Usuário para Sistemas Federados 403

Apêndice D. Opções de Coluna para Sistemas Federados 405

Apêndice E. Opções de Mapeamentos de Função para Sistemas Federados 407

Apêndice F. Tipos de Servidores Válidos em Instruções SQL. 409

| | |
|-------------------------|-----|
| Wrapper CTLIB | 409 |
|-------------------------|-----|

| | |
|------------------------------|-----|
| Wrapper DBLIB | 409 |
| Wrapper DJXMSSQL3 | 409 |
| Wrapper DRDA | 410 |
| Wrapper Informix. | 411 |
| Wrapper MSSQLODBC3 | 411 |
| Wrapper NET8. | 411 |
| Wrapper ODBC | 411 |
| Wrapper OLE DB. | 412 |
| Wrapper SQLNET | 412 |
| Wrapper Teradata. | 412 |

Apêndice G. Mapeamentos de Tipos de Dados de Avanço Padrão 413

| | |
|---|-----|
| Origens de Dados DB2 para z/OS e OS/390 | 414 |
| Origens de Dados DB2 para iSeries | 415 |
| Origens de Dados DB2 Server para VM e VSE | 417 |
| Origens de Dados DB2 para Linux, UNIX e Windows | 418 |
| Origens de Dados Informix | 420 |
| Origens de Dados Microsoft SQL Server | 422 |
| Origens de Dados ODBC | 425 |
| Origens de Dados Oracle NET8 | 427 |
| Origens de Dados Oracle SQLNET | 428 |
| Origens de Dados Sybase | 429 |
| Origens de Dados Teradata | 431 |

Apêndice H. Mapeamentos de Tipos de Dados Reversos Padrão 433

| | |
|--|-----|
| Origens de Dados do DB2 para z/OS e OS/390 | 434 |
| Origens de Dados do DB2 para iSeries. | 435 |
| Origens de Dados do DB2 para VM e VSE | 436 |
| Origens de Dados do DB2 para Linux, UNIX e Windows | 437 |
| Origens de Dados do Informix | 438 |
| Origens de Dados Microsoft SQL Server | 440 |
| Origens de Dados do Oracle SQLNET | 441 |
| Origens de Dados do Oracle NET8 | 442 |
| Origens de Dados do Sybase | 443 |
| Origens de Dados do Teradata | 444 |

Documentação Técnica do DB2 Information Integrator. 447

| | |
|---|-----|
| Acessando Manuais e Informações sobre Release | 447 |
| Manuais do DB2 Information Integrator | 447 |
| Notas sobre o Release e Requisitos de Instalação | 450 |
| Fix Packs para a Documentação do DB2 Information Integrator | 450 |
| Acessando Tópicos Utilizando o DB2 Information Integrator Information Center ou o CD de Documentação em HTML do DB2 | 451 |
| Recursos do DB2 Information Integrator Information Center | 451 |
| Localizando Tópicos no DB2 Information Integrator Information Center. | 451 |
| Utilizando a Documentação em HTML do DB2 | 453 |
| Pesquisando a Documentação do DB2 | 456 |
| Pesquisando Documentações do DB2 sobre Resolução de Problemas com o Netscape 4.x | 457 |

Acessibilidade 459

| | |
|---|-----|
| Entrada de Dados e Navegação através do Teclado | 459 |
| Exibição Acessível | 459 |
| Definições das Fontes | 459 |
| Não-dependência de Cores. | 459 |
| Dicas de Alertas Alternativos | 460 |
| Compatibilidade com Tecnologias Assistidas | 460 |
| Documentação Acessível | 460 |

Avisos 461

| | |
|-----------------------------|-----|
| Marcas Comerciais | 463 |
|-----------------------------|-----|

Índice Remissivo 465

Entrando em Contato com a IBM. 471

| | |
|---|-----|
| Informações sobre o Produto | 471 |
| Comentários sobre a Documentação. | 471 |

Sobre este Manual

Este manual contém:

- Instruções para incluir origens de dados em um sistema federado, registrando invólucros. Invólucros são módulos que permitem que você ou um aplicativo se comuniquem com uma origem de dados, utilizando instruções SQL.

As alterações técnicas no texto são indicadas por uma linha vertical à esquerda da alteração.

Quem Deve Ler este Manual

Este manual é destinado a administradores que estejam configurando um ambiente de banco de dados federado e a programadores de aplicativos que estejam desenvolvendo aplicativos para tal ambiente.

Convenções

Este manual utiliza estas convenções para destaque:

Negrito

Indica comandos e controles da interface gráfica com o usuário (graphical user interface - GUI) (por exemplo, nomes de campos, nomes de pastas, opções de menu).

Espaçamento fixo

Indica exemplos de codificação do texto digitado.

Itálico Indica as variáveis que devem ser substituídas por um valor. O itálico também indica títulos de manuais e enfatiza palavras.

MAIÚSCULAS

Indica as palavras-chave e nomes de objetos SQL (por exemplo, tabelas, exibições e servidores).

Como Ler Diagramas de Sintaxe

Neste manual, a sintaxe é descrita usando-se a estrutura definida a seguir:

Leia os diagramas de sintaxes da esquerda para a direita e de cima para baixo, seguindo a direção da linha.

O símbolo ►— indica o início de uma instrução.

O símbolo \longrightarrow indica que a sintaxe da instrução continua na linha seguinte.

O símbolo \blacktriangleright indica que uma instrução continua a partir da linha anterior.

O símbolo $\longrightarrow\blacktriangleleft$ indica o fim de uma instrução.

Os itens obrigatórios aparecem na linha horizontal (no caminho principal).



Os itens opcionais aparecem abaixo do caminho principal.

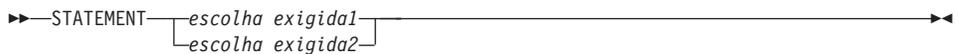


Se um item opcional aparecer acima do caminho principal, ele não terá efeito na execução da instrução e será usado somente para leitura.



Se você puder escolher dentre dois ou mais itens, eles aparecerão em pilhas.

Se você *tiver* que escolher um dos itens, um item da pilha aparecerá no caminho principal.



Se não escolher nenhum dos itens for uma opção, a pilha inteira aparecerá abaixo do caminho principal.



Se um dos itens for o padrão, aparecerá acima do caminho principal e as opções restantes serão mostradas abaixo.



Uma seta voltada para a esquerda, acima da linha principal, indica um item que pode ser repetido. Neste caso, os itens repetidos devem ser separados por um ou mais espaços em branco.



Se a seta repetida contiver uma vírgula, você deverá separar os itens repetidos com uma vírgula.



Uma seta repetida acima de uma pilha indica que você pode fazer mais de uma opção dos itens empilhados ou repetir uma única opção.

As palavras-chave aparecem em maiúsculas (por exemplo, FROM). Elas devem ser digitadas exatamente como mostradas. As variáveis aparecem em minúsculas (por exemplo, nome-da-coluna). Elas representam nomes ou valores fornecidos pelo usuário na sintaxe.

Caso pontos de acentuação, parênteses, operadores aritméticos ou outros símbolos forem mostrados, você deverá digitá-los como parte da sintaxe.

Algumas vezes, uma única variável representa um conjunto de vários parâmetros. Por exemplo, no diagrama a seguir, a variável `bloco-do-parâmetro` pode ser substituída por qualquer interpretação do diagrama que é o principal **bloco-do-parâmetro**:



bloco-do-parâmetro:



Os segmentos adjacentes que ocorrem entre “marcadores grandes” (●) podem ser especificados em qualquer seqüência.



O diagrama acima mostra que item2 e item3 podem ser especificados em qualquer ordem. As duas opções a seguir são válidas:

```
STATEMENT item1 item2 item3 item4
```

```
STATEMENT item1 item3 item2 item4
```

Capítulo 1. Visão Geral sobre Como Configurar o Acesso às Origens de Dados

As seções a seguir fornecem um guia conciso sobre como configurar um servidor federado e um banco de dados para acessar suas origens de dados:

- Elas contêm informações sobre as etapas básicas necessárias para executar rapidamente as etapas de configuração.
- Elas descrevem várias etapas opcionais, se precisar delas, para ajustar bem a configuração da origem de dados.

Há capítulos individuais de configuração para cada origem de dados.

Trilha Rápida para Configurar Origens de Dados

Você pode executar a maior parte das etapas requeridas para configurar o acesso a uma origem de dados através do Centro de Controle do DB2®. Utilize o Centro de Comandos do DB2 para as etapas que requerem uma linha de comandos. Comute entre estas interfaces gráficas com o usuário para configurar rapidamente o acesso a uma origem de dados. As etapas para configurar o acesso são semelhantes, independentemente da origem de dados. As etapas básicas e a interface recomendada são:

Tabela 1. A interface recomendada e etapas de configuração

| Etapa de configuração | Interface recomendada | Notas |
|--|---------------------------------------|---|
| 1. Preparar o servidor federado para a origem de dados | Assistente de Configuração do Cliente | Para origens de dados da família DB2: Catalogar o nó e o banco de dados remoto Para origens de dados Informix, Oracle, Sybase, Microsoft® SQL Server: Configurar e testar o arquivo de configuração do cliente |
| 2. Criar os wrappers | Centro de Controle do DB2 | |

Tabela 1. A interface recomendada e etapas de configuração (continuação)

| Étapa de configuração | Interface recomendada | Notas |
|---|---------------------------|---|
| 3. Criar definições de servidor | Centro de Controle do DB2 | O conceito de um nó varia de origem de dados para origem de dados. Para origens de dados relacionais, um nó reflete uma instância do servidor da origem de dados. No DB2, um <i>nó</i> é equivalente a uma instância, que está executando a cópia do DB2. |
| 4. Criar mapeamentos de usuários | Centro de Controle do DB2 | Se você tentar recuperar a senha remota associada a um mapeamento de usuário a partir da exibição do catálogo SYSCAT.USEROPTIONS, o valor da senha remota será exibido criptografado. |
| 5. Testar a conexão com o servidor da origem de dados | Centro de Comando DB2 | Utilizar o painel Mostrar todas as Tabelas no Centro de Controle do DB2 para verificar as conexões. |
| 6. Criar pseudônimos | Centro de Controle do DB2 | |

No entanto, antes de configurar o acesso a uma origem de dados, é necessário certificar-se de que o servidor federado tenha sido configurado corretamente. É muito importante:

- Vincular o DB2 ao software cliente. Isto cria bibliotecas do wrapper da origem de dados no servidor federado.
- Configurar as variáveis de ambiente da origem de dados.

Related concepts:

- “Etapas de Configuração Opcionais” na página 8

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*

Related reference:

- “Regras de Nomenclatura de Objetos Federados” na página 19

Origens de Dados Suportadas

Há muitas origens de dados que podem ser acessadas utilizando um sistema federado. A tabela a seguir lista as origens de dados suportadas:

Tabela 2. Versões das origens de dados suportadas e métodos de acesso.

| Origem de dados | Versões suportadas | Método de acesso |
|---|--|--|
| DB2 Universal Database™ para Linux, UNIX e Windows® | 7.1, 7.2, 8.1 | DRDA® |
| DB2 Universal Database para z/OS™ e OS/390® | 6.1, 7.1 com os seguintes APARs aplicados: <ul style="list-style-type: none">• PQ62695• PQ55393• PQ56616• PQ54605• PQ46183• PQ62139 | DRDA |
| DB2 Universal Database para iSeries™ | 4.5 (ou posterior) com os seguintes APARs aplicados: <ul style="list-style-type: none">• SA95719• SE06003• SE06872• SI05990• SI05991 | DRDA |
| DB2 Server para VM e VSE | 7.1 (ou posterior) com correções para APARs para funções de esquema aplicadas. | DRDA |
| Informix™ | 7, 8, 9 | Informix Client SDK |
| ODBC | 3.x | Driver ODBC para a origem de dados, como o Driver Redbrick ODBC para acessar o Redbrick. |
| OLE DB | | OLE DB 2.0 (ou posterior) |
| Oracle | 7.3.4, 8.x, 9.x | Software cliente SQLNET ou NET8 |

Tabela 2. Versões das origens de dados suportadas e métodos de acesso. (continuação)

| Origem de dados | Versões suportadas | Método de acesso |
|----------------------------------|--|--|
| Microsoft SQL Server | 6.5, 7.0, 2000 | No Windows, o driver Microsoft SQL Server Client ODBC 3.0 (ou posterior). No UNIX, o driver DataDirect Technologies (anteriormente MERANT) Connect ODBC 3.7 (ou posterior). |
| Sybase | 11.x,12.x | Sybase Open Client |
| Teradata | V2R3, V2R4 | Teradata Call-Level Interface Versão 2 (CLIV2) Release 04.06 (ou posterior) |
| BLAST | 2.x | Daemon BLAST (fornecido com o wrapper) |
| Documentum | Servidor Documentum: EDMS 98 (também referido como versão 3) e 4i. | Documentum Client API/Library |
| Entrez | 1.0 | None |
| HMMER | 2.2g | Daemon HMMER (fornecido com o wrapper) |
| IBM Lotus Extended Search | 4.0 | Extended Search Client Library (fornecido com o wrapper) |
| Microsoft Excel | 97, 2000 | Excel 97 ou 2000 instalado no servidor federado |
| Arquivos com estrutura de tabela | | None |
| XML | Especificação 1.0 | Nenhum |

Related concepts:

- “Data sources” no *Federated Systems Guide*

Criar Pseudônimos para cada Objeto de Origem de Dados

A tarefa de criar um pseudônimo geralmente é a mais envolvida das tarefas de configuração. Esta seção fornece um exemplo do que você tem que fazer para identificar candidatos para pseudônimos e para registrar um pseudônimo em um objeto de origem de dados federado.

Os objetos de origem de dados podem ser relacionais ou não relacionais. Exemplos de objetos de origem de dados relacionais são: tabelas do banco de dados, exibições e sinônimos (apenas Informix). Exemplos de objetos de origem de dados não relacionais são: algoritmos de pesquisa BLAST, objetos e tabelas registradas em um Documentum Docbase, arquivos do Microsoft[®] Excel (.xls), arquivos com estrutura de tabela (.txt) e arquivos de marcação XML.

Tabelas e exibições que residem no banco de dados federado são *objetos locais*. Não é necessário criar pseudônimos para estes objetos. Utilize o nome de objeto real em suas consultas.

Objetos remotos são:

- Tabelas e exibições em outra instância de banco de dados DB2[®] no servidor federado. É necessário criar pseudônimos para estes objetos.
- Objetos de origem de dados que residem em outra origem de dados, como: Oracle, Sybase, Documentum e ODBC. É necessário criar pseudônimos para estes objetos.

Ao enviar uma pedido distribuído para o servidor federado, o pedido faz referência a um objeto de origem de dados por seu pseudônimo. Os pseudônimos são mapeados para nomes de objetos específicos na origem de dados. Os mapeamentos eliminam a necessidade de qualificar os pseudônimos por nomes de origem de dados. A localização dos objetos da origem de dados é transparente para o aplicativo cliente ou usuário final. Pseudônimos não são nomes alternativos para objetos de origem de dados. Eles são ponteiros pelos quais o servidor faz referência a estes objetos.

Por exemplo, se você definir o pseudônimo *DEPT* para representar uma tabela do banco de dados Informix[™] chamada *NFX1.PERSON.DEPT*, a instrução `SELECT * FROM DEPT` será permitida a partir do servidor federado. No entanto, a instrução `SELECT * FROM NFX1.PERSON.DEPT` não é permitida.

Ao criar um pseudônimo para um objeto de origem de dados relacional, os dados do catálogo do servidor remoto são recuperados e armazenados no catálogo global federado.

Para origens de dados não relacionais, a forma de armazenamento das informações de origens de dados no catálogo global varia de origem de dados para origem de dados. As informações podem ser recuperadas do servidor remoto, ou talvez seja necessário incluir estas informações na instrução CREATE NICKNAME.

O Compilador SQL utiliza estes metadados para facilitar o acesso ao objeto da origem de dados. Por exemplo, suponha que um pseudônimo esteja definido para uma tabela com um índice. Os metadados fornecidos para o catálogo global são informações relacionadas ao índice, como o nome de cada coluna na chave do índice.

Para criar um pseudônimo, utilize o Centro de Controle do DB2. Você também pode emitir a instrução CREATE NICKNAME no Centro de Comandos do DB2 ou no CLP (processador da linha de comandos). Você pode definir mais de um pseudônimo para o mesmo objeto da origem de dados.

O exemplo a seguir mostra uma instrução CREATE NICKNAME:

```
CREATE NICKNAME SYBSALES FOR SYBSERVER."salesdata"."europe"
```

onde:

SYBSALES

É um pseudônimo único para a tabela ou a exibição Sybase.

Nota: O pseudônimo é um nome com duas partes — o esquema e o pseudônimo. Se você omitir o esquema ao criar o pseudônimo, o esquema do pseudônimo será authid do usuário que estiver criando o pseudônimo. Os pseudônimos podem conter 128 caracteres.

SYBSERVER."salesdata"."europe"

É um identificador de três partes para o objeto de origem de dados remoto.

- *SYBSERVER* é o nome atribuído ao servidor da origem de dados na instrução CREATE SERVER.
- *salesdata* é o nome do esquema remoto ao qual o objeto pertence. Este valor faz distinção entre letras maiúsculas e minúsculas.
- *europe* é o nome do objeto remoto que você deseja acessar. Este valor faz distinção entre letras maiúsculas e minúsculas.

Ao criar o pseudônimo, o servidor federado utilizará o pseudônimo para testar a conexão com a origem de dados. Ele tenta consultar o catálogo da origem de dados. Se a conexão não funcionar, você receberá uma mensagem de erro.

Incluindo Opções de Colunas ao Criar um Pseudônimo

Suponha que você deseja criar o pseudônimo INDSALES para uma tabela chamada INDONESIA_SALES. A tabela contém a coluna POSTAL_CODE com o tipo de dados CHAR. A coluna contém apenas caracteres numéricos. A origem de dados possui uma seqüência de intercalação que se difere da seqüência de intercalação do banco de dados federado. Geralmente, o servidor federado não classificaria esta coluna na origem de dados. No entanto, a coluna POSTAL_CODE contém apenas caracteres numéricos ('0','1',..., '9'). Você pode indicar isto, atribuindo um valor 'Y' à opção de coluna NUMERIC_STRING. Isto permite que o otimizador de consultas DB2 execute a classificação na origem de dados. Se a classificação for executada remotamente, será possível evitar a sobrecarga de conversão de dados no servidor federado. Para fornecer estas informações para o servidor federado, adicione a opção de coluna NUMERIC_STRING à instrução CREATE NICKNAME. Por exemplo:

```
CREATE NICKNAME INDSALES FOR SERVER44."sales"."INDONESIA_SALES"  
OPTIONS (POSTAL_CODE NUMERIC_STRING 'Y')
```

Para algumas origens de dados não relacionais, os wrappers não contêm os mapeamentos de tipos padrão. Se o wrapper não contiver os mapeamentos de tipos padrão, os tipos de dados correspondentes do DB2 para UNIX® e Windows® deverão ser especificados para cada coluna do objeto da origem de dados quando o pseudônimo for criado. Cada coluna deve ser mapeada para um campo ou coluna específico no objeto da origem de dados. Por exemplo:

```
CREATE NICKNAME DRUGDATA1  
(DCODE INTEGER, DRUG CHAR(20), MANUFACTURER CHAR(20))  
FOR SERVER biochem_lab  
OPTIONS (FILE_PATH '/usr/pat/DRUGDATA1.TXT',  
COLUMN_DELIMITER ',', KEY_COLUMN 'Dcode', VALIDATE_DATA_FILE 'Y')
```

Criando um Pseudônimo em um Pseudônimo

Às vezes, pode ser necessário criar um pseudônimo em um pseudônimo. Suponha que você tenha um servidor federado utilizando o AIX® e um servidor federado utilizando o Windows. Você deseja acessar uma planilha do Excel a partir dos dois servidores federados. No entanto, o wrapper Excel é suportado apenas em servidores federados que utilizam o Windows. Para acessar a planilha do Excel a partir do servidor federado do AIX, siga estas etapas:

1. No servidor federado do Windows, instale e configure o servidor para acessar origens de dados do Excel.
2. Crie um pseudônimo para a planilha do Excel.
3. No servidor federado do AIX, instale e configure o servidor para acessar origens de dados da família DB2.
4. Crie um pseudônimo para o pseudônimo do Excel no servidor federado do Windows.

Related tasks:

- “Nicknames : Federated Systems help”
- “Filtering tables and views for creating nicknames : Federated Systems help”
- “Filtering tables for creating nicknames: Federated Systems help”
- “Creating nicknames: Federated Systems help”

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Regras de Nomenclatura de Objetos Federados” na página 19

Etapas de Configuração Opcionais

Você pode personalizar ou ajustar colunas de um pseudônimo que são utilizadas em consultas federadas da seguinte forma:

- Especifique índices para objetos que não possuem um índice quando você configurou originalmente o acesso à origem de dados. Por exemplo, você cria uma especificação de índice quando uma tabela adquire um novo índice. Da mesma forma, você cria uma especificação de índice se o objeto da origem de dados (como uma exibição) geralmente não possui índices.
- Defina mapeamentos de tipos de dados alternativos em vez de utilizar os mapeamentos de tipos de dados padrão. Você pode especificar um mapeamento que é utilizado apenas para um objeto de origem de dados específico, como uma tabela específica de um banco de dados.
- Defina mapeamentos de funções alternativas em vez de utilizar os mapeamentos de funções padrão. Isto é útil principalmente quando você deseja forçar o DB2® a utilizar uma função definida pelo usuário na origem de dados.

Capítulo 2. Visão Geral de um Sistema Federado

As seções a seguir fornecem uma visão geral de um sistema federado.

Wrappers e Módulos de Wrappers

Wrappers são mecanismos pelos quais o servidor federado interage com origens de dados. O servidor federado utiliza rotinas armazenadas em uma biblioteca chamada *módulo de wrapper* para implementar um wrapper. Essa rotinas permitem que o servidor federado execute operações como conectar-se a uma origem de dados e recuperar dados dela continuamente. Geralmente, o proprietário da instância federada do DB2[®] utiliza a instrução CREATE WRAPPER para registrar um wrapper no banco de dados federado.

Cria-se um wrapper para cada tipo de origem de dados que você deseje acessar. Por exemplo, suponhamos que você deseje acessar três tabelas do banco de dados DB2 para z/OS[™], uma tabela do DB2 para iSeries[™], duas tabelas do Informix[™] e uma exibição do Informix. É necessário criar apenas dois wrappers, um para os objetos da origem de dados DB2 e um para os objetos da origem de dados Informix. Depois desses wrappers estarem registrados no banco de dados federado, você pode utilizar esses wrappers para acessar outros objetos dessas origens de dados. Por exemplo, você pode utilizar o wrapper DRDA[®] com todos os objetos da origem de dados da família DB2 — DB2 para Linux, UNIX[®] e Windows, DB2 para z/OS e OS/390, DB2 para iSeries e DB2 Server para VM e VSE.

As definições e os pseudônimos do servidor são utilizados para identificar as especificações (nome, localização e assim por diante) de cada objeto da origem de dados.

Um wrapper executa muitas tarefas. Algumas dessas tarefas são:

- Conectar-se à origem de dados. O wrapper utiliza a API da conexão padrão da origem de dados.
- Enviar consultas à origem de dados.
 - Para as origens de dados que suportam SQL, a consulta é enviada em SQL.
 - Para as origens de dados que não suportam SQL, a consulta é convertida na linguagem de consulta nativa da origem ou em uma série de chamadas API da origem.
- Receber conjuntos de resultados da origem de dados. O wrapper utiliza as APIs padrão da origem de dados para receber o conjunto de resultados.

- Responder às consultas do servidor federado sobre os mapeamentos de tipo de dados padrão para uma origem de dados. O wrapper contém os mapeamentos de tipo padrão que são utilizados quando os pseudônimos são criados para um objeto da origem de dados. Para os wrappers relacionais, os mapeamentos de tipo de dados criados substituem os mapeamentos de tipo de dados padrão. Os mapeamentos de tipo de dados definidos pelo usuário são armazenados em um catálogo global.
- Responder às consultas do servidor federado sobre os mapeamentos de função padrão para uma origem de dados. O wrapper contém informações de que o servidor federado precisa para determinar se as funções do DB2 são mapeadas para funções da origem de dados e como as funções são mapeadas. Estas informações são utilizadas pelo SQL Compiler para determinar se a origem de dados é capaz de executar as operações de consulta. Para os wrappers relacionais, os mapeamentos de função criados substituem os mapeamentos de tipo de função padrão. Os mapeamentos de função definidos pelo usuário são armazenados em um catálogo global.

As *opções de wrapper* são utilizadas para configurar o wrapper ou para definir como o DB2 utiliza o wrapper.

Related concepts:

- “Definições do Servidor e Opções do Servidor” na página 10

Related reference:

- “Nomes de Wrappers Padrão” no *Federated Systems Guide*

Definições do Servidor e Opções do Servidor

Depois dos wrappers serem criados para as origens de dados, o proprietário da instância federada define as origens de dados para o banco de dados federado. O proprietário da instância fornece um nome para identificar a origem de dados e outras informações pertinentes à origem de dados. Se a origem de dados for um RDBMS, essas informações incluem:

- O tipo e versão do RDBMS
- O nome do banco de dados para a origem de dados no RDBMS
- Os metadados específicos do RDBMS

Por exemplo, uma origem de dados da família DB2[®] pode ter vários bancos de dados. A definição deve especificar a qual banco de dados o servidor federado pode conectar-se. Por outro lado, uma origem de dados do Oracle[®] tem um banco de dados e o servidor federado pode conectar-se ao banco de dados sem saber seu nome. O nome do banco de dados não está incluído na definição do servidor federado de uma origem de dados do Oracle.

O nome e outras informações que o proprietário da instância fornece ao servidor federado são coletivamente chamadas de *definição do servidor*. As origens de dados respondem pedidos de dados e são servidores por direito.

As instruções CREATE SERVER e ALTER SERVER são utilizadas para criar e modificar uma definição de servidor.

Algumas das informações de uma definição de servidor são armazenadas como *opções do servidor*. Ao criar definições de servidor, é importante compreender as opções que podem ser especificadas sobre o servidor. Algumas opções do servidor configuram o wrapper e algumas afetam a maneira pela qual o DB2 utiliza o wrapper.

As opções do servidor podem ser definidas para persistir durante conexões sucessivas da origem de dados ou durante uma única conexão.

Related concepts:

- “Mapeamentos de Usuário” na página 11

Related reference:

- Apêndice B, “Opções do Servidor para Sistemas Federados”, na página 391

Intercalando Seqüências e Configuração de Origem de Dados

Como parte da instalação do DB2 Information Integrator, o banco de dados federado foi criado. Nesse momento, uma seqüência de intercalação que corresponde à origem de dados foi designada. Ao registrar a definição de servidor no banco de dados federado, é necessário definir a opção de servidor COLLATING_SEQUENCE como ‘Y’. Esta definição instrui o banco de dados federado que as seqüências de intercalação do banco de dados federado e do servidor da origem de dados são correspondentes.

Mapeamentos de Usuário

Quando um servidor federado precisa efetuar pushdown de um pedido para uma origem de dados, o servidor primeiro deve estabelecer uma conexão com a origem de dados.

Para a maioria das origens de dados, o servidor federado faz isso utilizando um ID de usuário e uma senha válidos para a origem de dados. Quando um ID de usuário e uma senha são requeridos para conectar a uma origem de dados, você deve definir uma associação entre o ID de usuário e a senha do servidor federado e o ID de usuário e a senha da origem de dados. Essa

associação deve ser criada para cada ID do usuário que utilizará o sistema federado para enviar solicitações distribuídas. Essa associação chama-se *mapeamento do usuário*.

Related concepts:

- “Pseudônimos e Objetos da Origem de Dados” na página 12

Pseudônimos e Objetos da Origem de Dados

Depois de criar definições de servidor e mapeamentos de usuário, o proprietário da instância federada cria os pseudônimos. Um *pseudônimo* é um identificador que é utilizado para fazer referência ao objeto localizado nas origens de dados que você deseja acessar. Os objetos identificados por pseudônimos são referidos como *objetos de origem de dados*.

Pseudônimos não são nomes alternativos para objetos da origem de dados como os aliases são nomes alternativos. Eles são ponteiros pelos quais o servidor faz referência a estes objetos. Os pseudônimos são geralmente definidos com a instrução CREATE NICKNAME.

Quando um usuário final ou um aplicativo cliente envia um pedido distribuído ao servidor federado, o pedido não precisa especificar as origens de dados. Em vez disso, o pedido faz referência aos objetos da origem de dados por seu pseudônimo. Os pseudônimos são mapeados para objetos específicos na origem de dados. Esses mapeamentos eliminam a necessidade de qualificar os pseudônimos por nomes de origem de dados. A localização dos objetos da origem de dados é transparente para o usuário final ou para o aplicativo cliente.

Suponhamos que você defina o pseudônimo *DEPT* para representar uma tabela do banco de dados do Informix™ chamada *NFX1.PERSON*. A instrução `SELECT * FROM DEPT` é permitida a partir do servidor federado. No entanto, a instrução `SELECT * FROM NFX1.PERSON` não é permitida a partir do servidor federado (a não ser em uma sessão de passagem).

Ao criar um pseudônimo para um objeto da origem de dados, metadados sobre o objeto são adicionados ao catálogo global. O otimizador de consulta utiliza estes metadados e as informações do wrapper para facilitar o acesso ao objeto da origem de dados. Por exemplo, se o pseudônimo for para uma tabela que tem um índice, o catálogo global conterá informações sobre o índice. O wrapper contém os mapeamentos entre os tipos de dados do DB2® e os tipos de dados da origem de dados.

Atualmente, você não pode executar algumas operações de utilitário do DB2 (REORG, REORGCHK, IMPORT, RUNSTATS e assim por diante) em pseudônimos.

Related concepts:

- “Opções da Coluna” na página 14

Related reference:

- “Objetos Válidos da Origem de Dados” na página 13

Objetos Válidos da Origem de Dados

Os pseudônimos identificam na origem de dados objetos que você deseja acessar. A tabela a seguir lista os tipos de objetos para os quais é possível criar um pseudônimo em um sistema federado.

Tabela 3. Objetos válidos da origem de dados

| Origem de dados | Objetos válidos |
|--------------------------------|---|
| DB2 para Linux, UNIX e Windows | Pseudônimos, tabelas de consulta materializadas, tabelas, exibições |
| DB2 para z/OS e OS/390 | Tabelas, exibições |
| DB2 para iSeries | Tabelas, exibições |
| DB2 para VM e VSE | Tabelas, exibições |
| Informix | Tabelas, exibições, sinônimos |
| Microsoft SQL Server | Tabelas, exibições |
| ODBC | Tabelas, exibições |
| Oracle | Tabelas, exibições, sinônimos |
| Sybase | Tabelas, exibições |
| Teradata | Tabelas, exibições |
| BLAST | Arquivos FASTA indexados para algoritmos de pesquisa BLAST |
| Documentum | Objetos e tabelas registradas em um Documentum Docbase |
| Entrez | Bancos de dados Entrez |
| Pesquisa Estendida | Arquivos das origens de dados, como bancos de dados Lotus Notes, Microsoft Access, Microsoft Index Server, mecanismos de pesquisa da Web e diretórios LDAP. |

Tabela 3. Objetos válidos da origem de dados (continuação)

| Origem de dados | Objetos válidos |
|----------------------------------|---|
| HMMER | Arquivos de banco de dados HMM (bibliotecas HMM [Hierarchical Markov Models], como PFAM), que podem ser pesquisados pelo programa hmmpfam do HMMER. |
| Microsoft Excel | Arquivos .xls (apenas a primeira planilha da pasta de trabalho é acessada) |
| Arquivos com estrutura de tabela | Arquivos .txt (arquivos de texto que atendem a um formato muito específico) |
| Arquivos com tag XML | Conjuntos de itens de um documento XML |

Related concepts:

- “Pseudônimos e Objetos da Origem de Dados” na página 12
- “Opções da Coluna” na página 14

Opções da Coluna

Você pode fornecer ao catálogo global informações adicionais de metadados sobre o objeto com o pseudônimo. Estes metadados descrevem valores em determinadas colunas do objeto da origem de dados. Você atribui estes metadados a parâmetros chamados *opções da coluna*. As opções da coluna indicam ao wrapper como tratar os dados de uma coluna de forma diferente da que normalmente faria. O SQL Compiler e o otimizador de consulta utiliza os metadados para desenvolver melhores planos para acessar os dados.

As opções da coluna são utilizadas para fornecer outras informações ao wrapper também. Por exemplo, para as origens de dados XML, uma opção de coluna é utilizada para indicar ao wrapper a expressão XPath a ser utilizada quando o wrapper analisa a coluna fora do documento XML.

Com federação, o servidor DB2® trata o objeto da origem de dados ao qual um pseudônimo faz referência como se fosse uma tabela local do DB2. Como resultado, é possível definir as opções da coluna para qualquer objeto da origem de dados para o qual um pseudônimo é criado. Algumas opções da coluna são desenvolvidas para tipos específicos de origens de dados e podem ser aplicadas apenas a essas origens de dados.

Suponhamos que uma origem de dados possua uma seqüência de intercalação diferente da seqüência de intercalação do banco de dados federado. O servidor federado geralmente não ordena nenhuma coluna que contenha

dados de caracteres na origem de dados. Ele geralmente retorna os dados ao banco de dados federado e executa a classificação localmente. No entanto, suponhamos que a coluna seja do tipo de dados de caracteres (CHAR ou VARCHAR) e contenha apenas caracteres numéricos ('0','1',..., '9'). Você pode indicar isto, atribuindo um valor 'Y' à opção de coluna NUMERIC_STRING. Isto permite que o otimizador de consultas DB2 execute a classificação na origem de dados. Se a classificação for executada remotamente, é possível evitar a sobrecarga de conversão de dados no servidor federado e de execução da classificação localmente.

Atenção: A opção de coluna NUMERIC_STRING é válida apenas para origens de dados relacionais.

É possível definir opções da coluna nas instruções CREATE NICKNAME e ALTER NICKNAME.

Related concepts:

- “Mapeamentos de Tipos de Dados” na página 15

Related tasks:

- “Working with nicknames” no *Federated Systems Guide*

Related reference:

- Apêndice D, “Opções de Coluna para Sistemas Federados”, na página 405

Mapeamentos de Tipos de Dados

Os tipos de dados da origem de dados devem mapear para tipos de dados do DB2[®] correspondentes de forma que o servidor federado possa recuperar dados das origens de dados. Alguns exemplos de mapeamentos de tipo de dados padrão são:

- O tipo Oracle[®] FLOAT mapeia para o tipo DB2 DOUBLE
- O tipo Oracle DATE mapeia para o tipo DB2 TIMESTAMP
- O tipo DB2 para z/OS[™] DATE mapeia para o tipo DB2 DATE

Para a maioria das origens de dados, os mapeamentos de tipo padrão estão em wrappers. Os mapeamentos de tipo padrão para as origens de dados DB2 estão no wrapper DRDA[®]. Os mapeamentos de tipo padrão para o Informix[™] estão no wrapper INFORMIX, e assim por diante.

Para algumas origens de dados não-relacionais, você deve especificar as informações do tipo de dados na instrução CREATE NICKNAME. Os tipos de dados do DB2 para Linux, UNIX[®] e Windows[®] correspondentes devem ser especificados para cada coluna do objeto da origem de dados quando o

pseudônimo é criado. Cada coluna deve ser mapeada para um campo ou coluna específico no objeto da origem de dados.

Para as origens de dados relacionais, é possível substituir os mapeamentos de tipo de dados padrão ou criar mapeamentos quando não houver padrão. Por exemplo, é possível criar um mapeamento de tipo quando um novo tipo interno está disponível na origem de dados ou quando há um tipo definido pelo usuário na origem de dados para a qual deseja mapear.

Atenção: Você deve criar novos mapeamentos de tipo ou modificar os mapeamentos de tipo padrão antes de criar pseudônimos. Caso contrário, os pseudônimos criados antes das alterações de mapeamento de tipo não refletirão os novos mapeamentos.

Related concepts:

- “Mapeamentos de Tipos de Dados em um Sistema Federado” no *Federated Systems Guide*

Mapeamentos de Função

Para o servidor federado reconhecer uma função da origem de dados, a função deve estar mapeada contra uma função complementar existente no DB2® para Linux, UNIX® e Windows. O DB2 Information Integrator fornece mapeamentos padrão entre funções internas existentes da origem de dados e funções internas complementares do DB2. Para a maioria das origens de dados, os mapeamentos de funções padrão estão nos wrappers. Os mapeamentos de função padrão para as funções do DB2 para z/OS™ e OS/390® estão no wrapper DRDA®. Os mapeamentos de função padrão para as funções Sybase estão nos wrappers CTLIB e DBLIB e assim por diante.

Para as origens de dados relacionais, você pode criar um mapeamento de função quando quiser utilizar uma função de origem de dados que o servidor federado não reconhece. O mapeamento criado é entre a função da origem de dados e uma função complementar do DB2 do banco de dados federado. Os mapeamentos de função são geralmente utilizados quando uma nova função interna ou uma nova função definida pelo usuário torna-se disponível na origem de dados. Os mapeamentos de funções também são utilizados quando uma função de contraparte do DB2 não existe.

Related concepts:

- “Mapeamentos de Funções em um Sistema Federado” no *Federated Systems Guide*
- “Especificações do Índice” na página 17

Especificações do Índice

Ao criar um pseudônimo para uma tabela da origem de dados, as informações sobre qualquer índice que a tabela da origem de dados tiver serão adicionadas ao catálogo global. O otimizador de consulta utiliza essas informações para expedir o processamento de pedidos distribuídos. As informações do catálogo sobre um índice da origem de dados são um conjunto de metadados e são chamadas de *especificação do índice*. Um servidor federado não cria uma especificação de índice quando um pseudônimo é criado para:

- Uma tabela que não tenha índices
- Uma exibição que, geralmente, não tem nenhuma informação sobre o índice armazenada no catálogo remoto
- Um objeto da origem de dados que não tem um catálogo remoto a partir do qual o servidor federado possa obter as informações sobre o índice

Suponhamos que uma tabela adquira um novo índice, além dos que possuía quando o pseudônimo foi criado. Como as informações sobre o índice são fornecidas ao catálogo global na hora da criação do pseudônimo, o servidor federado não estará ciente do novo índice. De forma similar, quando um pseudônimo é criado para uma exibição, o servidor federado não está ciente da tabela subjacente (e seus índices) a partir da qual a exibição foi gerada. Nessas circunstâncias, você pode fornecer as informações de índice necessárias para o catálogo global. É possível criar uma especificação de índice para tabelas que não têm índices. A especificação de índice indica ao otimizador de consulta qual coluna ou quais colunas da tabela pesquisar para localizar dados rapidamente.

Related concepts:

- “Especificações de Índice em um Sistema Federado” no *Federated Systems Guide*

Capítulo 3. Planejando a Configuração da Origem de Dados Federada

As seções a seguir fornecem informações que podem ser utilizadas para ajudá-lo a planejar seu sistema federado.

Regras de Nomenclatura de Objetos Federados

Assim como outros objetos do DB2, existem regras de nomenclatura de objetos do banco de dados federado.

Os objetos do banco de dados federado incluem:

- Mapeamentos de funções
- Especificações de índice
- Pseudônimos
- Servidores
- Mapeamentos de tipos
- Mapeamentos de usuário
- Wrappers

Os nomes de objetos federados devem começar com um dos seguintes itens:

- Uma letra, incluindo uma letra acentuada válida (como Ö)
- Um caractere multibyte, exceto um espaço multibyte (para ambientes multibyte)

Os nomes de objetos federados não podem começar com um número ou com o caractere sublinhado.

Os nomes de objetos federados também podem incluir os seguintes caracteres:

- A a Z
- 0 a 9
- @, #, \$, e _ (sublinhado)

Os nomes de objetos federados não podem exceder 128 bytes.

As opções (como opções de servidor e opções de pseudônimo) e as definições de opções estão limitadas a 255 bytes.

Nomes sem aspas são convertidos em maiúscula.

Related concepts:

- “Naming rules in an NLS environment” no *Administration Guide: Planning*
- “Naming rules in a Unicode environment” no *Administration Guide: Planning*

Related reference:

- “Preservando Valores com Distinção entre Maiúsculas e Minúsculas em um Sistema Federado” na página 20

Preservando Valores com Distinção entre Maiúsculas e Minúsculas em um Sistema Federado

Em um sistema federado, às vezes é necessário especificar valores, como IDs do usuário e senhas, que fazem distinção entre maiúsculas e minúsculas na origem de dados. Para assegurar que o tipo de letra esteja correto quando os valores forem transmitidos para a origem de dados, siga estas diretrizes:

- Especifique os valores com o tipo de letra requerido e coloque-os corretamente entre aspas. Aspas duplas são opcionais para nomes de objetos, como o nome de um wrapper ou pseudônimo. Aspas simples são requeridas para valores de opções, como REMOTE_AUTHID e REMOTE_PASSWORD.
- Para IDs do usuário e senhas, você pode definir as opções de servidor FOLD_ID e FOLD_PW server para converterem automaticamente os valores no tipo de letra apropriado. Com esta opção, não é necessário lembrar o tipo de letra requerido para cada origem de dados. Você pode digitar os valores em qualquer tipo de letra e eles serão convertidos automaticamente.

Informações sobre opções de servidor e suas definições válidas são discutidas em tópicos separados.

A partir de um Prompt de Comandos do Sistema Operacional UNIX:

Se você colocar um valor com distinção entre maiúsculas e minúsculas entre aspas no prompt de comandos do sistema operacional do servidor federado, deverá assegurar que as aspas sejam analisadas corretamente:

- Suponha que a instrução SQL contenha aspas duplas, mas não contenha aspas simples. Em seguida, você coloca a instrução entre aspas simples. Por exemplo, se desejar emitir esta instrução SQL:

```
CREATE NICKNAME my_nick FOR my_server."owner"."my_table"
```

Insira o seguinte texto no prompt de comandos do UNIX

```
DB2 'CREATE NICKNAME my_nick FOR my_server."owner"."my_table"'
```

- Suponha que a instrução SQL contenha aspas simples, mas não contenha aspas duplas. Em seguida, você coloca 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')
```

Insira o seguinte texto no prompt de comandos do UNIX

```
DB2 "CREATE USER MAPPING FOR USER SERVER my_server
    OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password') "
```

- Suponha que a instrução SQL contenha aspas simples e duplas, em seguida, você coloca a instrução entre aspas duplas e precede todas as aspas duplas na instrução com uma barra invertida. Por exemplo, se desejar emitir esta instrução SQL:

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

Insira o seguinte texto no prompt de comandos do UNIX

```
DB2 "CREATE USER MAPPING FOR \"local_id\" SERVER my_server
    OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')"
```

Nota: Os exemplos acima assumem que você esteja inserindo instruções SQL a partir do prompt de comandos do UNIX e transmitindo a instrução para o comando db2, sem a opção -f. Se você inserir as instruções SQL a partir de um arquivo utilizando o comando db2 com a opção -f, não deverá preceder as aspas duplas com uma barra invertida.

A partir de um Prompt de Comandos do Sistema Operacional Windows:

No Windows, preceda cada aspa com uma barra invertida. Por exemplo, suponha que você deseja criar o pseudônimo NICK1 para uma tabela do Microsoft SQL Server. A tabela reside no banco de dados NORBASE. O esquema é *my_schema* e a tabela é *weekly_salary*.

No prompt de comandos do Windows, no servidor federado, digite:

```
DB2 CREATE NICKNAME nick1
    FOR norbase.\"my_schema\".\"weekly_salary\"
```

A partir do DB2 CLP ou de um Programa Aplicativo:

Ao inserir o valor a partir do CLP (prompt da linha de comandos) do DB2 ou ao especificar o valor em um programa aplicativo, não são necessárias as aspas simples ou as barras invertidas. Utilizando o exemplo acima, no prompt de comandos do DB2, digite:

```
CREATE NICKNAME nick1
    FOR norbase.\"my_schema\".\"weekly_salary\"
```

Related reference:

- Apêndice B, “Opções do Servidor para Sistemas Federados”, na página 391

Atualizar Estatísticas de Origem de Dados

O banco de dados federado depende das estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento da consulta. Essas estatísticas são coletadas quando você cria um pseudônimo para um objeto da origem de dados utilizando a instrução `CREATE NICKNAME`. O banco de dados federado verifica a presença do objeto na origem de dados e tenta coletar os dados estatísticos existentes da origem de dados. As informações úteis ao otimizador de consultas são lidas dos catálogos de origem de dados e colocadas no catálogo global no servidor federado. Como algumas informações do catálogo de origem de dados, ou todas elas, podem ser utilizadas pelo otimizador de consultas, é recomendável atualizar as estatísticas (utilizando o comando de origem de dados equivalente a `RUNSTATS`) na origem de dados antes de criar um pseudônimo.

O banco de dados federado recupera informações estatísticas remotas para um objeto apenas uma vez quando você cria um pseudônimo para o objeto. Se a origem remota atualizar suas estatísticas de catálogo para um objeto remoto depois de criar o pseudônimo, as informações estatísticas alteradas não serão propagadas para o catálogo global do servidor federado. Para certificar-se de que o catálogo global no servidor federado reflita as estatísticas atuais para o objeto remoto, é necessário eliminar e recriar o pseudônimo.

Ação: Identifique os objetos nas origens de dados que deseja incluir no servidor federado. Estes serão objetos para os quais você criará pseudônimos. Decida para qual destas origens de dados você pode atualizar as estatísticas e liste essas origens de dados na tabela de estatísticas da origem de dados na lista de verificação de planejamento.

Planejar Mapeamentos de Tipos de Dados

Os tipos de dados de origem de dados são referidos como tipos de dados *remotos* e os tipos de dados do banco de dados federado são referidos como tipos de dados *locais*.

Existem dois tipos de mapeamentos entre tipos de dados de origem de dados e tipos de dados do banco de dados federado: mapeamentos de tipos de avanço e mapeamentos de tipos reversos. Em um *mapeamento de tipo de avanço*, o mapeamento ocorre de um tipo remoto para um tipo local comparável. Um *mapeamento de tipo reverso* é utilizado com o DDL transparente. Em um mapeamento de tipo reverso, o mapeamento ocorre de um tipo local para um

tipo remoto comparável. Informações adicionais sobre os dois tipos de mapeamentos de tipos de dados são discutidas em tópicos separados.

O DB2 para UNIX e Windows utiliza mapeamentos de tipos de dados para determinar quais tipos de dados suportados pelo DB2 devem ser definidos para colunas em um objeto de origem de dados. Os mapeamentos de tipos de dados padrão são baseados nos wrappers de origem de dados.

No entanto, seus aplicativos podem requerer mapeamentos de tipos de dados que são diferentes dos mapeamentos padrão. Você pode substituir os mapeamentos padrão para:

- Alterar um mapeamento de tipo para todos os objetos de origem de dados localizados em um servidor específico
- Alterar um mapeamento de tipo para um objeto de origem de dados específico
- Alterar um mapeamento de tipo para um tipo de origem de dados específico
- Alterar um mapeamento de tipo para um tipo e versão de origem de dados específicos

Utilize a instrução `CREATE TYPE MAPPING` para definir novos mapeamentos de tipos de dados. Os mapeamentos criados estão armazenados na exibição `SYSCAT.TYPEMAPPINGS` do catálogo global do banco de dados federado.

Altere um mapeamento de tipo de dados *antes* de criar pseudônimos para os objetos de origem de dados. Quando você cria um pseudônimo para um objeto de origem de dados, o servidor federado ocupa o catálogo global com informações sobre a tabela. Estas informações incluem o pseudônimo, o nome da tabela da origem de dados, os nomes de colunas e os tipos de dados que estão definidos para cada coluna da tabela.

Somente os pseudônimos criados após a alteração de um mapeamento refletem o novo mapeamento de tipo. Os pseudônimos criados antes da alteração do mapeamento utilizarão o mapeamento de tipo de dados padrão.

Se você criar os mapeamentos de tipos de dados depois de criar os pseudônimos, será necessário alterar cada pseudônimo para refletir o novo mapeamento ou eliminar e recriar os pseudônimos.

Nota: Se uma tabela de origem de dados contiver colunas que sejam de tipos de dados distintos ou definidos pelo usuário, você terá duas opções:

- Você pode criar o mapeamento de tipo no banco de dados federado antes de criar um pseudônimo para essa tabela de origem de dados. Ao criar os mapeamentos de tipos antes de criar o pseudônimo, o servidor federado

saberá para qual tipo de dados mapeará estas colunas. Se os mapeamentos para estes tipos de dados distintos ou definidos pelo usuário não forem criados antes de você emitir a instrução CREATE NICKNAME, aparecerá um erro.

- Se as colunas na tabela de origem de dados atenderem qualquer uma das seguintes condições:
 - As colunas são tipos de dados definidos pelo usuário que são baseados no sistema ou em tipos de dados internos
 - As colunas possuem atributos que não são suportados para mapeamentos de tipos de dados

Você pode criar uma exibição na origem de dados na qual as colunas são associadas ou *lançadas* no tipo de dados interno subjacente. Em seguida, crie um pseudônimo para a exibição em vez de para a tabela.

Ação: Identifique os mapeamentos de tipos de dados para os quais deseja definir novos mapeamentos. Liste as origens de dados e os mapeamentos de tipos que deseja criar na tabela de mapeamentos de tipos de dados na lista de verificação de planejamento.

Related concepts:

- “Mapeamentos de Tipos de Dados” na página 15
- “Tuning query processing” no *Federated Systems Guide*

Related reference:

- Apêndice G, “Mapeamentos de Tipos de Dados de Avanço Padrão”, na página 413
- Apêndice H, “Mapeamentos de Tipos de Dados Reversos Padrão”, na página 433

Planejar Mapeamentos de Funções

O DB2 para UNIX e Windows fornece mapeamentos de funções padrão entre funções de origem de dados internas existentes e funções internas do DB2. Para a maioria das origens de dados, os mapeamentos de funções padrão estão nos wrappers. Para algumas origens de dados não relacionais, não é possível alterar os mapeamentos de funções padrão.

Para utilizar uma função de origem de dados que o servidor federado não reconhece, é necessário criar um mapeamento de funções. O mapeamento criado está entre a função da origem de dados e uma função de contraparte no banco de dados federado. Os mapeamentos de funções geralmente são utilizados quando uma nova função interna ou uma nova função definida pelo usuário se torna disponível na origem de dados.

Os mapeamentos de funções também são utilizados quando uma função de contraparte do DB2 não existe. Neste caso, antes de criar o mapeamento de funções, será necessário criar um gabarito de função no banco de dados federado.

Ação: Determine se você precisa criar mapeamentos de funções para suas origens de dados. Liste os mapeamentos de funções necessários na tabela de mapeamentos de funções na lista de verificação de planejamento.

Related concepts:

- “Mapeamentos de Função” na página 16

Planejar Mapeamentos do Usuário

Quando um servidor federado precisa efetuar pushdown de um pedido para uma origem de dados, o servidor primeiro deve estabelecer uma conexão com a origem de dados. O servidor faz isto utilizando um ID do usuário e uma senha válidos para essa origem de dados. Você deve definir uma associação entre o ID do usuário do servidor federado e da origem de dados. Essa associação deve ser criada para cada ID do usuário que utilizará o sistema federado para enviar solicitações distribuídas. Essa associação chama-se *mapeamento do usuário*.

Você pode utilizar o Centro de Controle do DB2 para criar um mapeamento de usuário para um grupo de usuários que acessarão uma origem de dados com os mesmos ID do usuário e senha.

Ação: Identifique os IDs do usuário que requerem um mapeamento de usuário entre o servidor federado e a origem de dados. Liste os IDs do usuário do servidor federado e os IDs do usuário da origem de dados correspondente na tabela de mapeamento de usuário na lista de verificação de planejamento.

Escolher o Wrapper Correto

Algumas origens de dados possuem mais de um wrapper que você pode utilizar. O escolhido depende da versão do software de clientes da origem de dados que você está utilizando. Ou, ele pode depender do sistema operacional existente em seu servidor federado.

Por exemplo, existem dois wrappers que podem ser utilizados com origens de dados Oracle: o wrapper SQLNET e o wrapper NET8. Suponha que esteja utilizando o Oracle Versão 8 e o sistema operacional no servidor federado seja o Windows NT. Originalmente, você cria o wrapper SQLNET. Posteriormente,

you perceive that the SQLNET wrapper does not support LOB data types, but the NET8 wrapper supports LOBs. To take advantage of LOB support in the NET8 wrapper, it will be necessary to eliminate the SQLNET wrapper and create the NET8 wrapper.

Nota: O wrapper NET8 requer uma versão do cliente Oracle mais recente do que a do wrapper SQLNET.

Existem importantes conseqüências em cascata quando você elimina um wrapper. Outros objetos no sistema federado são impactados:

- Todas as definições de servidor, mapeamentos de funções definidas pelo usuário e mapeamentos de tipos de dados definidos pelo usuário que são dependentes do wrapper eliminado também são eliminados.
- Todos os mapeamentos de funções definidos pelo usuário, pseudônimos, mapeamentos de tipos de dados definidos pelo usuário e mapeamentos de usuários que são dependentes da definição de servidor também são eliminados.
- As especificações de índice dependentes dos pseudônimos eliminados também são eliminadas.
- As exibições federadas dependentes destes pseudônimos são marcadas como inoperantes.
- Todos os aplicativos dependentes de objetos eliminados e exibições inoperantes são invalidados.

O DB2 Relational Connect fornece vários wrappers para origens de dados Oracle, Microsoft SQL Server e Sybase. As distinções entre os wrappers são discutidas nos tópicos de configuração para cada origem de dados.

Ação: Identifique os wrappers que serão criados para o sistema federado na tabela de wrappers na lista de verificação de planejamento.

Related concepts:

- “Wrappers e Módulos de Wrappers” na página 9

Related tasks:

- “Adding Microsoft SQL Server data sources to a federated server” no *Federated Systems Guide*
- “Adding Oracle data sources to a federated server” no *Federated Systems Guide*
- “Adding Sybase data sources to a federated server” no *Federated Systems Guide*

Lista de Verificação para Planejamento de Configuração do Sistema Federado

Você pode facilitar a configuração do sistema federado, seguindo esta lista de verificação de planejamento. Esta lista de verificação o orienta de várias maneiras para otimizar a configuração do sistema federado.

Lista de Verificação: Regras de Nomenclatura de Objetos Federados

Você está familiarizado com as regras de nomenclatura para objetos federados?

Consulte os links relacionados no final desta seção para localizar informações sobre as regras de nomenclatura de objetos federados.

Lista de Verificação: Preservando Valores com Distinção entre Maiúsculas e Minúsculas

Deseja definir as opções de servidor FOLD_ID e FOLD_PW para preservar o tipo de letra para os valores de ID do usuário e senha para as origens de dados? Utilize a tabela a seguir para identificar a quais definições de servidor estas opções serão aplicadas.

Tabela 4. Lista de verificação de planejamento: Opções de servidor FOLD_ID e FOLD_PW a serem definidas para o sistema federado

| Origem de dados | Nome do servidor (na definição de servidor) | Definição da opção de servidor FOLD_ID | Definição para a opção de servidor FOLD_PW |
|-----------------|---|--|--|
| | | | |
| | | | |
| | | | |
| | | | |

Lista de Verificação: Estatísticas de Origem de Dados

Na tabela a seguir, liste as origens de dados que farão parte de seu sistema federado. Indique para quais origens de dados você atualizará estatísticas.

Tabela 5. Lista de verificação de planejamento: Estatísticas de origens de dados a serem atualizadas para o sistema federado

| Origem de dados | Mantém informações do catálogo? (S/N) | Atualizará estatísticas para esta origem de dados? (S/N) | Nome do utilitário de origem de dados utilizado para atualizar estatísticas |
|-------------------------|---------------------------------------|--|---|
| DB2 para UNIX e Windows | S | S | RUNSTATS |
| | | | |

Tabela 5. Lista de verificação de planejamento: Estatísticas de origens de dados a serem atualizadas para o sistema federado (continuação)

| Origem de dados | Mantém informações do catálogo? (S/N) | Atualizará estatísticas para esta origem de dados? (S/N) | Nome do usuário de origem de dados utilizado para atualizar estatísticas |
|-----------------|---------------------------------------|--|--|
| | | | |
| | | | |
| | | | |

Lista de Verificação: Mapeamentos de Tipos de Dados

Na tabela a seguir, identifique os tipos de dados de origem de dados e os tipos de dados do servidor federado correspondentes para os quais é necessário criar um mapeamento.

Tabela 6. Lista de verificação de planejamento: Mapeamentos de tipos de dados a serem criados para o sistema federado

| Origem de dados | Nome do servidor (na definição de servidor) | Tipo de dados da origem de dados | Tipo de dados do DB2 para UNIX e Windows |
|-----------------|---|----------------------------------|--|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Lista de Verificação: Mapeamentos de Usuários

Na tabela a seguir, identifique os IDs do usuário do servidor federado e os IDs do usuário correspondentes para *cada* origem de dados que fará parte do sistema federado.

Tabela 7. Lista de verificação de planejamento: Mapeamentos de usuários a serem criados para o sistema federado

| | | Origem de dados | Origem de dados | Origem de dados |
|-----------------|--|-----------------|-----------------|-----------------|
| | | _____ | _____ | _____ |
| Nome do usuário | ID do usuário do DB2 para UNIX e Windows | ID de usuário | ID de usuário | ID de usuário |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Lista de Verificação: Wrappers

Na tabela a seguir, identifique os wrappers que serão criados.

Tabela 8. Lista de verificação de planejamento: Wrappers a serem criados para o sistema federado

| Origem de dados | Nomes de wrappers padrão | Wrapper a ser criado |
|---|--------------------------|----------------------|
| DB2 Universal Database™ para UNIX e Windows® DB2 Universal Database para z/OS e OS/390® DB2 Universal Database para iSeries DB2 Server para VM e VSE | DRDA | |
| Informix | INFORMIX | |
| Oracle | SQLNet Net8 | |
| Microsoft® SQL Server | DJXMSSQL3 MSSQLODBC3 | |
| ODBC | nenhum | |
| OLE DB | OLEDB | |
| Sybase | CTLIB DBLIB | |

Tabela 8. Lista de verificação de planejamento: Wrappers a serem criados para o sistema federado (continuação)

| Origem de dados | Nomes de wrappers padrão | Wrapper a ser criado |
|----------------------------------|---------------------------------|-----------------------------|
| BLAST | nenhum | |
| Documentum | nenhum | |
| Microsoft Excel | nenhum | |
| Arquivos com estrutura de tabela | nenhum | |
| XML | nenhum | |

Capítulo 4. Configurando o Acesso às Origens de Dados da Família DB2

Este capítulo explica como configurar seu servidor federado para acessar dados armazenados em bancos de dados da família DB2. Esses bancos de dados incluem o DB2 para UNIX e Windows, o DB2 para z/OS e OS/390, o DB2 para iSeries e o DB2 Server para VM e VSE.

Este capítulo lista as tarefas que precisam ser executadas, mostra exemplos das instruções SQL que precisam ser emitidas e fornece informações sobre ajuste e resolução de problemas que você pode utilizar ao fazer a configuração das origens de dados da família DB2.

Adicionando Origens de Dados da Família DB2 a Servidores Federados

A configuração do servidor federado para acessar origens de dados DB2 envolve o fornecimento do servidor com informações sobre as origens de dados DB2 e objetos que você deseja acessar.

Você pode configurar o acesso a origens de dados DB2 através do Centro de Controle do DB2, através do Centro de Comandos do DB2 ou através do processador da linha de comandos do DB2.

A vantagem de utilizar o Centro de Controle do DB2 é que não é necessário digitar cada instrução e comando. O Centro de Controle do DB2 fornece o método mais fácil de configurar rapidamente o acesso a origens de dados DB2. No entanto, você não pode utilizar o Centro de Controle do DB2 para iniciar as seguintes tarefas de configuração:

- Catalogar o nó
- Catalogar o banco de dados remoto
- Testar a conexão com o servidor da origem de dados para validar a definição de servidor e os mapeamentos de usuários
- Adicionar ou eliminar opções de colunas

Pré-requisitos:

- Acesse o Centro de Comandos do DB2 ou o processador da linha de comandos do DB2
- Um servidor federado e um banco de dados que são configurados para acessar origens de dados da família DB2

Restrição:

Você não pode criar um pseudônimo para um alias da origem de dados DB2 se estiver acessando dados que estão armazenados no DB2 para UNIX e Windows, Versão 8.1.

Procedimento:

Para adicionar uma origem de dados DB2 a um servidor federado:

1. Catalogue o nó.
2. Catalogue o banco de dados remoto.
3. Registre o wrapper.
4. Registre a definição de servidor e defina as opções de servidor.
5. Crie os mapeamentos do usuário.
6. Teste a conexão com o servidor DB2.
7. Registre pseudônimos para tabelas e exibições.

Related concepts:

- “Trilha Rápida para Configurar Origens de Dados” na página 1

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Catalogando uma Entrada do Nó no Diretório do Nó Federado” na página 32
- “Ajustando e Resolvendo Problemas de Configuração para Origens de Dados da Família DB2” na página 44

Catalogando uma Entrada do Nó no Diretório do Nó Federado

A catalogação de uma entrada do nó no diretório do nó federado faz parte da maior tarefa de adicionar origens de dados da família DB2 a servidores federados.

Para apontar para a localização da origem de dados DB2, catalogue uma entrada no diretório do nó do servidor federado. O servidor federado utiliza esta entrada para determinar o método de acesso correto para conectar-se a uma origem de dados DB2.

Procedimento:

Para catalogar uma entrada do nó no diretório do nó federado:

1. Determine o protocolo de comunicação que será utilizado.
2. Emita o comando apropriado para catalogar a entrada do nó.
 - Se seu protocolo de comunicação for o TCP/IP (Transmission Control Protocol/Internet Protocol), emita o comando **CATALOG TCPIP NODE**.

Por exemplo:

```
CATALOG TCPIP NODE DB2NODE REMOTE SYSTEM42 SERVER DB2TCP42
```

O valor *DB2NODE* é o nome atribuído ao nó que está sendo catalogado. *REMOTE SYSTEM42* é o nome do host do sistema no qual a origem de dados reside. *SERVER DB2TCP42* é o nome do serviço ou número de porta primário da instância do gerenciador de banco de dados do servidor. Se for utilizado um nome de serviço, ele fará distinção entre maiúsculas e minúsculas.

- Se seu protocolo de comunicação for SNA, emita o comando **CATALOG APPC NODE**.

Por exemplo:

```
CATALOG APPC NODE DB2NODE REMOTE DB2CPIC SECURITY PROGRAM
```

O valor *DB2NODE* é o nome atribuído ao nó que está sendo catalogado. *REMOTE DB2CPIC* é o nome da LU (unidade lógica) parceira SNA do nó parceiro remoto. *SECURITY PROGRAM* especifica que um nome de usuário e uma senha devem ser incluídos no pedido de alocação que é enviado à LU parceira.

A próxima tarefa nesta seqüência de tarefas é catalogar o banco de dados remoto no diretório do banco de dados do sistema federado.

Related tasks:

- “Catalogando o Banco de Dados Remoto no Diretório do Banco de Dados do Sistema Federado” na página 33

Catalogando o Banco de Dados Remoto no Diretório do Banco de Dados do Sistema Federado

A catalogação do banco de dados remoto no diretório do banco de dados do sistema federado faz parte da maior tarefa de adicionar origens de dados da família DB2 a servidores federados.

Especifique a qual banco de dados da origem de dados DB2 o servidor federado se conecta, catalogando o banco de dados remoto no diretório do banco de dados do sistema do servidor federado.

Procedimento:

Para catalogar o banco de dados no diretório do banco de dados do sistema do servidor federado:

1. Utilize o CCA (Client Configuration Assistant).

Para servidores federados no UNIX, como alternativa, você pode utilizar o comando **CATALOG DATABASE**. Por exemplo:

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

O valor *DB2DB390* é o nome do banco de dados remoto que você está catalogando no diretório do banco de dados do sistema do servidor federado. *AS CLIENTS390* é o alias para o banco de dados que está sendo catalogado. Se não for especificado um alias, o gerenciador do banco de dados utilizará o nome do banco de dados (por exemplo, *DB2DB390*) como o alias. *AT NODE DB2NODE* é o nome do nó especificado ao catalogar a entrada do nó no diretório do nó. **AUTHENTICATION SERVER** especifica a autenticação que ocorre no nó da origem de dados DB2.

2. Se o nome do banco de dados remoto tiver mais de oito caracteres, será necessário criar uma entrada de diretório DCS, emitindo o comando **CATALOG DCS DATABASE**. Por exemplo:

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

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

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper DB2.

Related tasks:

- “Catalogando uma Entrada do Nó no Diretório do Nó Federado” na página 32
- “Registrando o Wrapper DB2” na página 34

Registrando o Wrapper DB2

O registro do wrapper DB2 faz parte da maior tarefa de adicionar origens de dados da família DB2 a servidores federados.

Para especificar o wrapper que será utilizado para acessar origens de dados DB2, emita a instrução **CREATE WRAPPER**. Cada DB2 Server Edition (Enterprise, Personal, e Workgroup) inclui um wrapper chamado DRDA para a família DB2.

Procedimento:

Para especificar o wrapper que deseja utilizar para acessar origens de dados da família DB2, emita a instrução CREATE WRAPPER.

Por exemplo:

```
CREATE WRAPPER DRDA
```

Recomendação: Utilize o nome do wrapper padrão chamado DRDA. Ao registrar o wrapper utilizando o nome padrão, o servidor federado utiliza automaticamente o nome da biblioteca padrão associado a esse nome de wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, será possível substituir o nome do wrapper padrão por um nome de sua escolha. Se você utilizar um nome que seja diferente do padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER. Suponha que você tenha um servidor federado em execução no AIX e decida utilizar um nome de wrapper diferente do padrão. É necessário incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.

Por exemplo:

```
CREATE WRAPPER mywrapper LIBRARY 'libdb2drda.a'
```

O valor *mywrapper* é o nome fornecido ao wrapper em vez de utilizar o nome do wrapper padrão.

A tabela a seguir lista os nomes de bibliotecas do wrapper para o DB2 por sistema operacional:

Tabela 9. Nomes de bibliotecas do wrapper DB2

| Sistema operacional no servidor federado | Nome da biblioteca do wrapper |
|--|-------------------------------|
| AIX | libdb2drda.a |
| Ambiente Operacional Solaris | libdb2drda.so |
| HP-UX | libdb2drda.sl |
| Linux | libdb2drda.so |
| Windows NT e Windows 2000 | db2drda.dll |

A próxima tarefa nesta seqüência de tarefas é registrar definições de servidor para uma origem de dados DB2.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados DB2” na página 36

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando Definições de Servidor para uma Origem de Dados DB2

O registro de definições de servidor para uma origem de dados DB2 faz parte da maior tarefa de adicionar origens de dados DB2 a servidores federados.

No banco de dados federado, você deve definir cada servidor DB2 que deseja acessar. Ao registrar a definição de servidor, o servidor federado conecta-se ao servidor DB2 e liga pacotes ao banco de dados. Como as informações para autorização e senha não são armazenadas no catálogo global federado, você deve incluí-las na definição de servidor.

Procedimento:

Para registrar uma definição de servidor para uma origem de dados DB2, emita a instrução CREATE SERVER.

Por exemplo:

```
CREATE SERVER server_name TYPE DB2/ZOS VERSION 6 WRAPPER DRDA
  AUTHORIZATION "name1" PASSWORD "passwd1"
  OPTIONS (DBNAME 'db_name')
```

O nome atribuído a um servidor deve ser exclusivo. Não são permitidos nomes de servidores duplicados.

A opção VERSION especificada é a versão do servidor do banco de dados DB2 que você deseja acessar. As versões suportadas são:

- DB2 para UNIX e Windows, Versão 6, Versão 7.1, Versão 7.2 e Versão 8.1
- DB2 para z/OS e OS/390, Versão 5 ou posterior
- DB2 para iSeries, Versão 4 ou posterior

O nome do parâmetro WRAPPER deve ser o nome especificado na instrução CREATE WRAPPER.

Embora o nome do banco de dados seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dados DB2.

Ao emitir a instrução CREATE SERVER, o servidor federado testará a conexão com o servidor da origem de dados DB2.

Depois de registrar a definição de servidor, você pode adicionar ou eliminar opções de servidor, emitindo a instrução ALTER SERVER.

A próxima tarefa nesta sequência de tarefas é criar o mapeamento de usuário para uma origem de dados DB2.

Related tasks:

- “Criando o Mapeamento de Usuário para uma Origem de Dados DB2” na página 38

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- “Instrução CREATE SERVER - Exemplos para o Wrapper DB2” na página 37

Instrução CREATE SERVER - Exemplos para o Wrapper DB2

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE SERVER para registrar servidores para wrappers em origens de dados da família DB2. Este tópico inclui um exemplo completo, que mostra como criar um servidor com todos os parâmetros obrigatórios e um exemplo com opções de servidor adicionais.

Exemplo completo:

O exemplo a seguir mostra como criar uma definição de servidor para um wrapper DB2 utilizando a instrução CREATE SERVER:

```
CREATE SERVER DB2SERVER TYPE DB2/ZOS VERSION 6 WRAPPER DRDA
AUTHORIZATION "spalten" PASSWORD "db2guru"
OPTIONS (DBNAME 'CLIENTS390')
```

DB2SERVER

Um nome atribuído ao servidor do banco de dados DB2. Esse nome deve ser exclusivo. Não são permitidos nomes de servidores duplicados.

TYPE DB2/ZOS

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

VERSION 6

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

WRAPPER DRDA

O nome especificado na instrução CREATE WRAPPER.

AUTHORIZATION "spalten"

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

PASSWORD "db2guru"

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

DBNAME 'CLIENTS390'

O alias para o banco de dados DB2 que você deseja acessar. Você definiu este alias quando catalogou o banco de dados utilizando o comando **CATALOG DATABASE**. Este valor faz distinção entre letras maiúsculas e minúsculas.

Este nome de banco de dados é requerido para origem de dados DB2.

Exemplo de opção do servidor:

Ao registrar a definição de servidor, você pode especificar opções de servidores adicionais na instrução **CREATE SERVER**. Estas opções incluem opções gerais de servidor e opções de servidor específicas de origens de dados DB2.

O exemplo a seguir mostra uma definição de servidor com a opção **CPU_RATIO**.

```
CREATE SERVER DB2SERVER TYPE DB2/ZOS VERSION 6 WRAPPER DRDA
  AUTHORIZATION "spalten" PASSWORD "db2guru"
  OPTIONS (DBNAME 'CLIENTS390', CPU_RATIO '0.001')
```

Se você definir a opção **CPU_RATIO** como '0.001', isto indica a CPU na origem de dados remota com capacidade disponível 1000 vezes maior do que o servidor federado.

Related tasks:

- "Registrando Definições de Servidor para uma Origem de Dados DB2" na página 36

Related reference:

- "CREATE SERVER statement" no *SQL Reference, Volume 2*
- Apêndice F, "Tipos de Servidores Válidos em Instruções SQL", na página 409

Criando o Mapeamento de Usuário para uma Origem de Dados DB2

A criação do mapeamento de usuário para uma origem de dados DB2 faz parte da maior tarefa de adicionar origens de dados da família DB2 a servidores federados.

Ao tentar acessar um servidor DB2, o servidor federado estabelece uma conexão com a origem de dados utilizando um ID do usuário e senha válidos

para essa origem de dados. É necessário definir uma associação (mapeamento de usuário) entre o ID do usuário do servidor federado e o ID do usuário e senha da origem de dados correspondente. Crie um mapeamento de usuário para cada ID do usuário que acessará o sistema federado para enviar pedidos distribuídos.

Procedimento:

Para mapear o ID do usuário local para o ID do usuário e senha do servidor DB2, emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR USERID SERVER DB2SERVER
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

O REMOTE_AUTHID é o ID de autorização de conexão, não o ID de autorização de ligação.

A próxima tarefa nesta seqüência de tarefas é testar a conexão com o servidor da origem de dados DB2.

Related tasks:

- “Testando a Conexão com o Servidor da Origem de Dados DB2” na página 40

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Instrução CREATE USER MAPPING - Exemplos para o Wrapper DB2” na página 39

Instrução CREATE USER MAPPING - Exemplos para o Wrapper DB2

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE USER MAPPING para mapear um ID do usuário local para o ID do usuário e senha do servidor DB2. Este tópico inclui um exemplo completo com todos os parâmetros requeridos e um exemplo que mostra como utilizar o registro especial USER do DB2 com a instrução CREATE USER MAPPING.

Exemplo completo:

O exemplo a seguir mostra como mapear um ID do usuário local para o ID do usuário do servidor DB2:

```
CREATE USER MAPPING FOR DB2USER SERVER DB2SERVER
    OPTIONS (REMOTE_AUTHID 'db2admin', REMOTE_PASSWORD 'day2night')
```

DB2USER

Especifica o ID do usuário local que você está mapeando para um ID do usuário definido em um servidor de origem de dados da família DB2.

SERVER *DB2SERVER*

Especifica o nome da origem de dados da família DB2 definido na instrução CREATE SERVER.

REMOTE_AUTHID *'db2admin'*

Especifica o ID de autorização de conexão no servidor da origem de dados da família DB2 para o qual você está mapeando *DB2USER*. Utilize aspas simples para preservar para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_ID como 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD *'day2night'*

Especifica a senha associada a *'db2admin'*. Utilize aspas simples para preservar para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_PW como 'U' ou 'L' na instrução CREATE SERVER.

Exemplo especial de registro:

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

```
CREATE USER MAPPING FOR USER SERVER DB2SERVER
    OPTIONS (REMOTE_AUTHID 'db2admin', REMOTE_PASSWORD 'day2night')
```

Você pode utilizar o registro especial USER 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.

Related tasks:

- “Criando o Mapeamento de Usuário para uma Origem de Dados DB2” na página 38

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Testando a Conexão com o Servidor da Origem de Dados DB2

O teste da conexão com o servidor da origem de dados DB2 faz parte da maior tarefa de adicionar origens de dados da família DB2 a servidores federados.

Você pode testar a conexão com o servidor DB2 utilizando a definição de servidor e mapeamentos de usuários definidos.

Procedimento:

Para testar a conexão:

1. Abra uma sessão de passagem para emitir uma instrução SQL SELECT nas tabelas do sistema DB2.

Por exemplo:

- No DB2 para z/OS e OS/390:

```
SET PASSTHRU server_name
SELECT count(*) FROM sysibm.systables
SET PASSTHRU RESET
```

- No DB2 para iSeries:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM qsys2.systables
SET PASSTHRU RESET
```

Se a instrução SQL SELECT retornar uma contagem, a definição de servidor e o mapeamento do usuário estão configurados adequadamente.

2. Se a instrução SQL SELECT retornar um erro, é possível que seja necessário:
 - Verificar o servidor remoto para certificar-se de que ele tenha sido iniciado.
 - Verificar o atendente no servidor remoto para certificar-se de que ele esteja configurado para conexões de entrada.
 - Verificar o mapeamento de usuário para certificar-se de que as definições para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com o servidor DB2.
 - Verificar as entradas do catálogo do DB2 para o nó e o banco de dados.
 - Verificar as definições das variáveis federadas do DB2 para verificar se é possível acessar o servidor DB2 remoto. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis db2dj.ini e a variável DB2 Profile Registry (db2set) DB2COMM.
 - Verificar a definição de servidor. Se necessário, elimine a definição de servidor e recrie-a.
 - Verificar o mapeamento de usuário. Se necessário, altere o mapeamento de usuário ou crie outro.

A próxima etapa nesta seqüência de tarefas é registrar pseudônimos para tabelas e exibições do DB2.

Related concepts:

- “Definições do Servidor e Opções do Servidor” na página 10

Related tasks:

- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Registrando Pseudônimos para Tabelas e Exibições do DB2” na página 42

Related reference:

- “ALTER USER MAPPING statement” no *SQL Reference, Volume 2*

Registrando Pseudônimos para Tabelas e Exibições do DB2

O registro de pseudônimos para tabelas de exibições do DB2 faz parte da maior tarefa de adicionar origens de dados da família DB2 a servidores federados.

O banco de dados federado depende das estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento da consulta. Essas estatísticas são reunidas quando você registra um pseudônimo para um objeto da origem de dados utilizando a instrução `CREATE NICKNAME`. O banco de dados federado verifica a presença do objeto na origem de dados e tenta coletar os dados estatísticos existentes da origem de dados. As informações úteis ao otimizador são lidas dos catálogos de origem de dados e colocadas no catálogo global no servidor federado. Como algumas ou todas as informações do catálogo de origem de dados podem ser utilizadas pelo otimizador, é recomendável atualizar estatísticas (utilizando o comando da origem de dados equivalente ao comando `RUNSTATS`) na origem de dados antes de registrar um pseudônimo.

Utilize a instrução `CREATE NICKNAME` para registrar um pseudônimo para uma exibição ou tabela localizada na origem de dados da família DB2. Utilize estes pseudônimos, em vez dos nomes de objetos de origem de dados, quando consultar a origem de dados da família DB2.

Procedimento:

Para registrar um pseudônimo, emita a instrução `CREATE NICKNAME`.

Por exemplo:

```
CREATE NICKNAME DB2NICKNAME FOR DB2SERVER.remote_schema.remote_table
```

Os pseudônimos podem ter até 128 caracteres.

Repita esta etapa para cada tabela ou exibição do DB2 para a qual você deseja registrar um pseudônimo.

Ao registrar o pseudônimo, o servidor federado utilizará a conexão para consultar o catálogo de origem de dados. Esta consulta testa a conexão com a origem de dados utilizando o pseudônimo. Se a conexão não funcionar, você receberá uma mensagem de erro.

Related reference:

- “RUNSTATS Command” na publicação *Command Reference*
- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper DB2” na página 43

Instrução CREATE NICKNAME - Exemplos para o Wrapper DB2

Este tópico fornece um exemplo que mostra como utilizar a instrução CREATE NICKNAME para registrar um pseudônimo para uma tabela ou exibição do DB2 que você deseja acessar.

O exemplo a seguir mostra uma instrução CREATE NICKNAME:

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

DB2SALES

Um pseudônimo exclusivo que é utilizado para identificar a tabela ou exibição do DB2.

Nota: O pseudônimo é um nome com duas partes que inclui o esquema e o pseudônimo. Se omitir o esquema ao registrar o pseudônimo, o esquema do pseudônimo será o ID de autorização do usuário que está criando o pseudônimo.

DB2SERVER.SALESDATA.EUROPE

Um identificador de três partes do objeto remoto:

- *DB2SERVER* é o nome atribuído ao servidor do banco de dados DB2 na instrução CREATE SERVER.
- *SALESDATA* é o nome do esquema remoto ao qual a tabela ou exibição pertence. Este valor faz distinção entre letras maiúsculas e minúsculas.
- *EUROPE* é o nome da tabela ou exibição remota que você deseja acessar.

Ajustando e Resolvendo Problemas de Configuração para Origens de Dados da Família DB2

Depois de fazer a configuração para origens de dados DB2, talvez você queira modificar a configuração para aprimorar o desempenho. Por exemplo, talvez você queira definir a variável de registro do perfil DB2_DJ_COMM DB2 para aprimorar o desempenho quando a origem de dados DB2 for acessada.

Aprimorando o Desempenho, Definindo a Variável DB2_DJ_COMM (UNIX)

Se perceber que demora muito tempo para acessar o servidor da origem de dados DB2, você poderá aprimorar o desempenho, definindo a variável de registro do perfil DB2_DJ_COMM. Ao definir a variável DB2_DJ_COMM, o servidor federado carregará o wrapper durante a inicialização em vez de quando você tenta acessar a origem de dados.

Procedimento:

Para definir a variável DB2_DJ_COMM:

1. Defina a variável DB2_DJ_COMM para o wrapper da biblioteca correspondente ao wrapper especificado. Utilize os comandos na tabela a seguir para definir a variável DB2_DJ_COMM.

Tabela 10. Comandos para definir a variável DB2_DJ_COMM para origens de dados DB2

| Sistema operacional do servidor federado | Comando |
|--|------------------------------|
| AIX | DB2_DJ_COMM= 'libdb2drda.a' |
| Ambiente Operacional Solaris | DB2_DJ_COMM= 'libdb2drda.so' |
| HP-UX | DB2_DJ_COMM= 'libdb2drda.sl' |
| Linux | DB2_DJ_COMM= 'libdb2drda.so' |
| Windows NT e Windows 2000 | DB2_DJ_COMM= 'db2drda.dll' |

Utilize o comando **db2set** para definir a variável DB2_DJ_COMM. Por exemplo, se o sistema operacional do servidor federado for AIX, o comando será:

```
db2set DB2_DJ_COMM='libdb2drda.a'
```

2. Exporte a variável DB2_DJ_COMM. Por exemplo:

```
export DB2_DJ_COMM
```
3. Emita os comandos a seguir, para reciclar a instância do DB2:

```
db2stop  
db2start
```

Ao reciclar a instância do DB2, assegure que ela aceite alterações de variáveis feitas.

Related tasks:

- “Adicionando Origens de Dados da Família DB2 a Servidores Federados” na página 31

Related reference:

- “db2set - DB2 Profile Registry Command” em *Command Reference*

Capítulo 5. Configurando o Acesso às Origens de Dados do Informix

Este capítulo explica como configurar seu servidor federado para acessar dados armazenados em origens de dados do Informix.

Este capítulo lista as tarefas que precisam ser executadas, mostra exemplos das instruções SQL que precisam ser emitidas e fornece informações sobre ajuste e resolução de problemas que você pode utilizar ao fazer a configuração das origens de dados do Informix.

Adicionando Origens de Dados Informix a Servidores Federados

A configuração do servidor federado para acessar origens de dados Informix envolve o fornecimento de informações ao servidor sobre as origens de dados Informix e objetos que você deseja acessar.

Você pode configurar o acesso a origens de dados Informix através do Centro de Controle do DB2, através do Centro de Comandos do DB2 ou através do processador da linha de comandos do DB2.

A vantagem de utilizar o DB2 Control Center é que não é necessário digitar cada instrução e comando. O Centro de Controle do DB2 fornece o método mais fácil de configurar rapidamente o acesso a origens de dados Informix. No entanto, você não pode utilizar o Centro de Controle do DB2 para iniciar as seguintes tarefas de configuração:

- Definir e testar o arquivo de configuração do cliente Informix
- Testar a conexão com o servidor Informix para validar a definição do servidor e mapeamentos de usuários
- Incluir ou eliminar opções de coluna

Pré-requisitos:

- Acesse o DB2 Command Center ou o processador da linha de comandos do DB2.
- Um servidor federado e um banco de dados que são configurados para acessar origens de dados Informix.
- O software SDK do Cliente Informix que está instalado e configurado no servidor federado.
- A configuração apropriada de variáveis de ambiente do sistema, variáveis db2dj.ini (incluindo variáveis de conversão de página de código) e variáveis

DB2 Profile Registry (db2set). As variáveis são: INFORMIXDIR, INFORMIXSERVER, CLIENT_LOCALE (opcional), DB_LOCALE (opcional), DBNLS (opcional) e INFORMIXSQLHOSTS (opcional). Você deve definir a variável INFORMIXSQLHOSTS apenas se o arquivo ou registro sqlhosts não estiver na localização padrão.

- Em servidores federados AIX, o AIX Base Application Development Math Library. Você pode determinar se Library está instalado emitindo o comando do AIX `lspp -l bos.adt.libm`.

Procedimento:

Para adicionar uma origem de dados Informix a um servidor federado:

1. Configure e teste o arquivo de configuração do cliente Informix.
2. Registre o wrapper.
3. Registre a definição de servidor.
4. Crie os mapeamentos do usuário.
5. Teste a conexão com o servidor Informix.
6. Registre pseudônimos para tabelas, exibições e sinônimos Informix.

Related concepts:

- “Trilha Rápida para Configurar Origens de Dados” na página 1

Related tasks:

- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Definindo e Testando o Arquivo de Configuração do Cliente Informix” na página 48
- “Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Informix” na página 60

Definindo e Testando o Arquivo de Configuração do Cliente Informix

A definição e teste do arquivo de configuração do cliente Informix fazem parte da maior tarefa de adicionar origens de dados Informix a servidores federados.

O arquivo de configuração do cliente é utilizado para conectar-se ao Informix utilizando as bibliotecas de cliente que estão instaladas no servidor federado. Este arquivo especifica a localização de cada servidor do banco de dados Informix e o tipo de conexão (protocolo) para o servidor do banco de dados.

- Em sistemas operacionais UNIX, o nome padrão é \$INFORMIXDIR/etc/sqlhosts. O arquivo sqlhosts reside em cada instalação do SDK do cliente Informix.
- Em sistemas operacionais Windows, a localização padrão do registro sqlhosts é o computador local.

O formato de sqlhosts é descrito no *Administrator's Guide for Informix Dynamic Server*.

Procedimento:

Para definir e testar o arquivo de configuração do cliente Informix:

1. Crie o arquivo sqlhosts ou defina o registro com o utilitário Setnet32 Informix.

Você pode copiar o arquivo ou registro sqlhosts de outro sistema que tenha o Informix Connect ou o SDK do Informix Client instalado. Você também pode configurar o SDK do Informix Client no servidor federado para conectar-se a um servidor Informix, que cria o arquivo ou registro sqlhosts. O servidor federado utilizará o sqlhosts que está no diretório SDK do Informix ou no registro do Windows.
2. Verifique a localização do arquivo ou registro sqlhosts.
 - Em sistemas operacionais UNIX, o arquivo sqlhosts está localizado no diretório \$INFORMIXDIR/etc/sqlhosts.
 - Em sistemas operacionais Windows, as informações de sqlhosts são mantidas na seguinte chave no registro do Windows:
HKEY_LOCAL_MACHINE\SOFTWARE\INFORMIX\SQLHOSTS
3. Se o arquivo ou registro sqlhosts não estiver na localização padrão, defina a variável de ambiente INFORMIXSQLHOSTS.
 - a. Em sistemas operacionais UNIX, defina a variável de ambiente INFORMIXSQLHOSTS como o nome completo do arquivo sqlhosts. Em sistemas operacionais Windows, defina a variável de ambiente INFORMIXSQLHOSTS como o nome do computador do Windows que armazena o registro.
 - b. Emita os seguintes comandos para reciclar a instância do DB2 e para assegurar que a variável de ambiente esteja definida no programa:
db2stop
db2start
4. Teste a conexão para certificar-se de que o software de cliente possa conectar-se ao servidor Informix. Se a ferramenta **dbaccess** do Informix estiver no servidor federado, utilize-a para testar a conexão. De outra maneira, execute o programa Informix de demonstração para testar a instalação do cliente.

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper Informix.

Related tasks:

- “Registrando o Wrapper Informix” na página 50
- “Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Informix” na página 60

Registrando o Wrapper Informix

O registro do wrapper Informix faz parte da maior tarefa de adicionar origens de dados Informix a servidores federados.

Para especificar o wrapper que será utilizado para acessar as origens de dados Informix, utilize a instrução `CREATE WRAPPER`. Cada DB2 Server Edition (Enterprise, Personal, Workgroup) inclui um wrapper para Informix chamado `INFORMIX`.

Procedimento:

Para especificar o wrapper que deseja utilizar para acessar origens de dados Informix, emita a instrução `CREATE WRAPPER`.

Por exemplo:

```
CREATE WRAPPER INFORMIX
```

Recomendação: Utilize o nome do wrapper padrão `INFORMIX`. Ao registrar o wrapper utilizando o nome padrão, o servidor federado utiliza automaticamente o nome da biblioteca padrão associado a esse nome de wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, será possível substituir o nome do wrapper padrão por um nome de sua escolha. Se você utilizar um nome diferente de um dos nomes padrão, deverá incluir o parâmetro `LIBRARY` na instrução `CREATE WRAPPER`.

Suponha que você tenha um servidor federado em execução no AIX e decida utilizar um nome de wrapper que não seja um dos nomes padrão. É necessário incluir o parâmetro `LIBRARY` na instrução `CREATE WRAPPER`.

Por exemplo:

```
CREATE WRAPPER mywrapper LIBRARY 'libdb2informix.a'
```

Os nomes de biblioteca do wrapper para o Informix são:

Tabela 11. Nomes de bibliotecas do wrapper Informix

| Sistema operacional no servidor federado | Nome da biblioteca do wrapper |
|--|-------------------------------|
| AIX | libdb2informix.a |
| HP-UX | libdb2informix.sl |
| Linux | libdb2informix.so |
| Ambiente Operacional Solaris | libdb2informix.so |
| Windows NT e Windows 2000 | db2informix.dll |

A próxima tarefa nesta seqüência de tarefas é registrar definições de servidor para uma origem de dados Informix.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Informix” na página 51

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando Definições de Servidor para uma Origem de Dados Informix

O registro de definições de servidor para uma origem de dados Informix faz parte da maior tarefa de adicionar origens de dados Informix a servidores federados.

No banco de dados federado, você deve definir cada servidor Informix que deseja acessar. Primeiro, você deve localizar o nome do nó da origem de dados Informix e, em seguida, utilizar este nome do nó ao registrar o servidor.

Procedimento:

Para registrar uma definição de servidor para uma origem de dados Informix:

1. Localize o nome do nó no arquivo ou registro sqlhosts do Informix.

Exemplo de arquivo sqlhosts:

```
inf724 onsoctcp anaconda inmx724
inf731 onscotcp boa ifmx731
inf92 onsoctcp python ifmx92
```

O primeiro valor em cada linha é *node_name*, tal como, inf724.

O segundo valor em cada linha é *nettype* ou tipo de conexão. Neste exemplo, `onscotcp` indica que esta é uma conexão TCP/IP.

O terceiro valor em cada linha é o nome do host, tal como, `anaconda`, `boa` e `python`.

O quarto valor em cada linha é o nome do serviço, tal como, `inmx724`. O campo de nome do serviço depende do *nettype* listado no segundo valor.

Embora o *node_name* esteja especificado como uma opção na instrução `CREATE SERVER SQL`, ele é obrigatório para origens de dados do Informix.

Para obter mais informações sobre o formato deste arquivo e significado destes campos, consulte o manual do Informix *Administrators Guide for Informix Dynamic Server*.

2. Emita a instrução `CREATE SERVER`.

Por exemplo:

```
CREATE SERVER server_name TYPE informix VERSION 9 WRAPPER INFORMIX
      OPTIONS (NODE 'node_name', DBNAME 'db_name')
```

Após a definição de servidor ter sido criada, utilize a instrução `ALTER SERVER` para incluir ou eliminar opções do servidor.

A próxima tarefa nesta seqüência de tarefas é criar o mapeamento de usuário para uma origem de dados Informix.

Related tasks:

- “Criando o Mapeamento de Usuário para uma Origem de Dados Informix” na página 54

Related reference:

- “`CREATE SERVER` statement” no *SQL Reference, Volume 2*
- Apêndice B, “Opções do Servidor para Sistemas Federados”, na página 391
- “Instrução `CREATE SERVER` - Exemplos para o Wrapper Informix” na página 52

Instrução `CREATE SERVER` - Exemplos para o Wrapper Informix

Este tópico fornece vários exemplos que mostram como utilizar a instrução `CREATE SERVER` para registrar servidores para o wrapper Informix. Este tópico inclui um exemplo completo, que mostra como registrar um servidor com parâmetros requeridos e um exemplo com opções de servidor adicionais.

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 9 WRAPPER INFORMIX  
OPTIONS (NODE 'abc', DBNAME 'sales', IUD_APP_SVPT_ENFORCE 'N')
```

asia Um nome atribuído ao servidor do banco de dados Informix. Esse nome deve ser exclusivo. Não são permitidos nomes de servidores duplicados.

TYPE *informix*

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso. Para o wrapper Informix, o tipo de servidor deve ser *informix*.

VERSION 9

A versão do servidor do banco de dados Informix que você deseja acessar. As versões do Informix suportadas são 7, 8 e 9.

WRAPPER *INFORMIX*

O nome especificado na instrução CREATE WRAPPER.

NODE '*abc*'

O nome do nó no qual o servidor do banco de dados Informix reside. Obtenha o nome do nó a partir do arquivo *sqlhosts*. Este valor faz distinção entre letras maiúsculas e minúsculas.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dados Informix.

DBNAME '*sales*'

O nome do banco de dados Informix que você deseja acessar. Este valor faz distinção entre letras 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 Informix.

IUD_APP_SVPT_ENFORCE '*N*'

Especifica se o sistema federado DB2 deve aplicar a detecção ou construção de instruções de ponto de salvamento de aplicativos. O Informix não suporta instruções de ponto de salvamento de aplicativos. Quando definido como '*N*', o servidor federado permitirá instruções INSERT, UPDATE ou DELETE nos pseudônimos para origens de dados Informix.

A opção de servidor IUD_APP_SVPT_ENFORCE deve ser definida como '*N*' para ativar a replicação para ou a partir de origens de dados Informix.

Embora a aplicação do ponto de salvamento de aplicativos seja especificada como uma opção na instrução CREATE SERVER, ela é requerida para origens de dados Informix.

Exemplo de opções de servidor:

Ao criar a definição de servidor você pode especificar opções de servidores adicionais na instrução CREATE SERVER. Estas opções de servidor incluem opções gerais de servidor e opções de servidor específicas do Informix.

O exemplo a seguir mostra uma definição de servidor Informix com opções de servidor adicionais:

```
CREATE SERVER asia TYPE informix VERSION 9 WRAPPER INFORMIX
      OPTIONS (NODE 'abc', DBNAME 'sales', FOLD_ID 'N', FOLD_PW 'N')
```

Quando o servidor federado conecta-se a uma origem de dados, ele tenta conectar-se utilizando todas as combinações possíveis de letras maiúsculas e minúsculas para o ID do usuário e senha, bem como o tipo de letra atual. O servidor pode fazer até nove tentativas de conexão antes de conectar-se com êxito ao servidor da origem de dados. Estas tentativas podem prolongar os tempos de conexão e podem resultar no bloqueio do ID do usuário. Você pode evitar bloqueios, especificando valores para as opções de servidor FOLD_ID e FOLD_PW.

Por exemplo, você pode definir as opções de servidor FOLD_ID e FOLD_PW como 'N' (não dobre o ID do usuário ou senha). Se você estabelecer estas definições, deverá especificar o ID do usuário e senha no tipo de letra correto. A vantagem de definir estas opções como 'N' é que, quando for especificado um ID do usuário ou senha inválida, o wrapper não continuará tentando as várias combinações. Esta definição reduz a chance de exceder o número máximo de tentativas de login com falha e de bloqueio do ID.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Informix” na página 51

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Criando o Mapeamento de Usuário para uma Origem de Dados Informix

A criação do mapeamento de usuário para uma origem de dados Informix faz parte da maior tarefa de adicionar origens de dados Informix a servidores federados.

Ao tentar acessar um servidor Informix, o servidor federado estabelece uma conexão com a origem de dados utilizando um ID do usuário e senha válidos para essa origem de dados. É necessário definir uma associação (mapeamento do usuário) entre cada ID do usuário do servidor federado e o ID do usuário e senha da origem de dados correspondente. Crie um mapeamento de usuário para cada ID do usuário que acessará o sistema federado para enviar pedidos distribuídos para a origem de dados Informix.

Procedimento:

Para mapear um ID do usuário local para o ID do usuário e senha do servidor Informix, emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR USERID SERVER INFORMIXSERVER
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

A próxima tarefa nesta seqüência de tarefas é testar a conexão com o servidor Informix.

Related tasks:

- “Testando a Conexão com o Servidor Informix” na página 56

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Instrução CREATE USER MAPPING - Exemplos para o Wrapper Informix” na página 55

Instrução CREATE USER MAPPING - Exemplos para o Wrapper Informix

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE USER MAPPING para mapear um ID do usuário do servidor federado para um ID do usuário e senha do servidor Informix. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o registro especial do DB2 USER com a instrução CREATE USER MAPPING.

Exemplo completo:

O exemplo a seguir mostra como mapear um ID do usuário do servidor federado (*VINCENT*) para um ID do usuário e senha do servidor Informix (*'vinnie'* e *'close2call'*):

```
CREATE USER MAPPING FOR VINCENT SERVER asia
    OPTIONS (REMOTE_AUTHID 'vinnie', REMOTE_PASSWORD 'close2call')
```

VINCENT

Especifica o ID do usuário local que você está mapeando para um ID do usuário definido em um servidor Informix.

SERVER *asia*

Especifica o nome do servidor Informix registrado na instrução CREATE SERVER.

REMOTE_AUTHID '*vinnie*'

Especifica o ID do usuário no servidor do banco de dados Informix para o qual você está mapeando VINCENT. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_ID como 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD '*close2call*'

Especifica a senha associada a '*vinnie*'. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_PW como 'U' ou 'L' na instrução CREATE SERVER.

Exemplo especial de registro:

O exemplo a seguir mostra uma instrução CREATE USER MAPPING que inclui o registro especial USER:

```
CREATE USER MAPPING FOR USER SERVER asia
  OPTIONS (REMOTE_AUTHID 'vinnie', REMOTE_PASSWORD 'close2call')
```

Você pode utilizar o registro especial USER 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 REMOTE_AUTHID do usuário.

Related tasks:

- “Criando o Mapeamento de Usuário para uma Origem de Dados Informix” na página 54

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Testando a Conexão com o Servidor Informix

O teste da conexão com o servidor Informix faz parte da maior tarefa de adicionar origens de dados Informix a servidores federados.

Você pode testar a conexão com o servidor Informix utilizando a definição de servidor e mapeamentos de usuários definidos.

Procedimento:

Para testar a conexão:

1. Abra uma sessão de passagem para emitir uma instrução SQL SELECT nas tabelas do sistema Informix.

Por exemplo:

```
SET PASSTHRU server_name
SELECT count(*) FROM informix.systables
SET PASSTHRU RESET
```

Se a instrução SQL SELECT retornar uma contagem, a definição de servidor e o mapeamento do usuário estão configurados adequadamente.

2. Se a instrução SQL SELECT retornar um erro, é possível que seja necessário:
 - Verificar o servidor Informix para certificar-se de que ele esteja configurado para conexões de entrada.
 - Verificar o mapeamento de usuário para certificar-se de que as definições para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com o servidor Informix. Alterar o mapeamento de usuário ou criar outro mapeamento, conforme necessário.
 - Verificar o software SDK do Cliente Informix no servidor federado DB2 para certificar-se de que ele esteja instalado e configurado corretamente para conectar-se ao servidor Informix.
 - Verificar as definições das variáveis federadas do DB2 para verificar se estão corretas para o servidor Informix. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis db2dj.ini e a variável DB2 Profile Registry (db2set).
 - Verifique sua definição de servidor. Se necessário, elimine-o e crie-o novamente.

A próxima etapa nesta sequência de tarefas é registrar pseudônimos para tabelas, exibições e sinônimos do Informix.

Related concepts:

- “Definições do Servidor e Opções do Servidor” na página 10

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*

- “Registrando Pseudônimos para Tabelas, Exibições e Sinônimos do Informix” na página 58

Related reference:

- “ALTER USER MAPPING statement” no *SQL Reference, Volume 2*

Registrando Pseudônimos para Tabelas, Exibições e Sinônimos do Informix

O registro de pseudônimos para tabelas, exibições e sinônimos do Informix faz parte da maior tarefa de adicionar origens de dados Informix a servidores federados.

Para cada servidor Informix definido, registre um pseudônimo para cada tabela, exibição ou sinônimo que deseja acessar. Utilize estes pseudônimos, em vez dos nomes de objetos de origem de dados, quando consultar os servidores Informix.

O banco de dados federado depende das estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento da consulta. Essas estatísticas são coletadas ao registrar um pseudônimo para um objeto da origem de dados utilizando a instrução CREATE NICKNAME. O banco de dados federado verifica a presença do objeto na origem de dados e tenta coletar os dados estatísticos existentes da origem de dados. As informações úteis ao otimizador são lidas a partir dos catálogos de origem de dados e colocadas no catálogo global no servidor federado. Como algumas informações do catálogo de origem de dados, ou todas elas, podem ser utilizadas pelo otimizador, atualize as estatísticas (utilizando o comando de origem de dados equivalente ao comando RUNSTATS do DB2) na origem de dados antes de registrar um pseudônimo.

Procedimento:

Para registrar um pseudônimo, emita a instrução CREATE NICKNAME.

Por exemplo:

```
CREATE NICKNAME informix_name FOR INFOSERVER."remote_schema".remote.table"
```

Os pseudônimos podem ter até 128 caracteres.

Repita esta etapa para cada tabela, exibição ou sinônimo do Informix para o qual deseja criar um pseudônimo.

Quando você cria o pseudônimo, o DB2 utilizará a conexão para consultar o catálogo de origens de dados. Esta consulta testa a conexão com a origem de dados utilizando o pseudônimo. Se a conexão não funcionar, você receberá uma mensagem de erro.

Related reference:

- “RUNSTATS Command” na publicação *Command Reference*
- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper Informix” na página 59

Instrução CREATE NICKNAME - Exemplos para o Wrapper Informix

Este tópico fornece um exemplo que mostra como utilizar a instrução CREATE NICKNAME para registrar um pseudônimo para uma tabela, exibição ou sinônimo do Informix que você deseja acessar.

Este exemplo mostra como especificar um objeto remoto para o servidor Informix com o qual o pseudônimo é atribuído:

```
CREATE NICKNAME JPSALES FOR asia.salesdata.japan"
```

JPSALES

Um pseudônimo exclusivo utilizado para identificar a tabela, exibição ou sinônimo.

Observação: o pseudônimo é composto de um nome de duas partes — o esquema e o pseudônimo. Se omitir o esquema ao registrar o pseudônimo, o esquema do pseudônimo será o ID de autorização do usuário que está registrando o pseudônimo.

asia.*salesdata*.*japan*"

Um identificador de três partes para o objeto remoto.

- *asia* é o nome atribuído ao servidor do banco de dados Informix na instrução CREATE SERVER.
- *salesdata* é o nome do esquema remoto ao qual a tabela, exibição ou sinônimo pertence.
- *japan* é o nome da tabela remota, exibição ou sinônimo que você deseja acessar.

O servidor federado dobra nomes de esquemas e tabelas do Informix para maiúsculas, a menos que você coloque os nomes entre aspas.

Related tasks:

- “Registrando Pseudônimos para Tabelas, Exibições e Sinônimos do Informix” na página 58

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*

Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Informix

Depois de fazer a configuração para origens de dados Informix, talvez você queira modificar a configuração para aprimorar o desempenho. Por exemplo, você pode definir a variável do registro de perfil DB2_DJ_COMM para aprimorar o desempenho quando a origem de dados Informix for acessada.

Aprimorando o Desempenho Definindo as Opções de Servidor FOLD_ID e FOLD_PW

Quando o servidor federado conecta-se a uma origem de dados, ele tenta conectar-se utilizando todas as combinações possíveis de letras maiúsculas e minúsculas para o ID do usuário e senha, bem como o tipo de letra atual. O servidor pode fazer até nove tentativas de conexão antes de conectar-se com êxito ao servidor da origem de dados. Essas tentativas podem reduzir os tempos de conexão.

Procedimento:

Para aprimorar o desempenho, especifique valores para as opções de servidor FOLD_ID e FOLD_PW utilizando a instrução ALTER SERVER OPTION.

- Suponha que todos os IDs do usuário e senhas do Informix estejam em minúsculas e a definição de FOLD_ID e FOLD_PW como o valor L (delimitado por aspas simples) possa 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')
```

- Como o servidor federado tenta cada combinação de valores em maiúsculas e minúsculas para o ID do usuário e senha, você pode reduzir a chance de exceder o número máximo de falhas de tentativas de login e de bloqueio do ID, definindo estas opções como 'N' (não dobre o ID do usuário e a senha). Se você estabelecer estas definições, deverá sempre especificar o ID do usuário e senha no tipo de letra correto. Se forem especificados um ID do usuário e senha inválidos, o wrapper não continuará tentando as diversas combinações. Por exemplo:

```
ALTER SERVER TYPE INFORMIX OPTIONS (ADD FOLD_ID 'N')
ALTER SERVER TYPE INFORMIX OPTIONS (ADD FOLD_PW 'N')
```

Aprimorando o Desempenho Definindo a Variável DB2_DJ_COMM (UNIX)

Se perceber que demora muito tempo para acessar o servidor Informix, será possível aprimorar o desempenho, definindo a variável do registro de perfil

DB2_DJ_COMM DB2 para carregar o wrapper quando o servidor federado é inicializado em vez de quando você tentar acessar a origem de dados.

Procedimento:

Para definir a variável DB2_DJ_COMM:

1. Defina a variável DB2_DJ_COMM para o wrapper da biblioteca correspondente ao wrapper especificado. Suponha que o servidor federado esteja em execução no AIX. O comando para definir a variável DB2_DJ_COMM é:

```
db2set DB2_DJ_COMM='libdb2informix.a,libdb2informixF.a,libdb2informixU.a'
```

A tabela a seguir lista os comandos DB2_DJ_COMM com os nomes de bibliotecas apropriados por sistema operacional.

Tabela 12. Comandos para definir a variável DB2_DJ_COMM para origens de dados Informix

| Sistema operacional do servidor federado | Comando |
|--|----------------------------------|
| AIX | DB2_DJ_COMM= 'libdb2informix.a' |
| HP-UX | DB2_DJ_COMM= 'libdb2informix.sl' |
| Linux | DB2_DJ_COMM= 'libdb2informix.so' |
| Ambiente Operacional Solaris | DB2_DJ_COMM= 'libdb2informix.so' |

2. Emita os comandos a seguir, para reciclar a instância do DB2:

```
db2stop  
db2start
```

Ao reciclar a instância do DB2, assegure que ela aceite alterações de variáveis feitas

Related tasks:

- “Adicionando Origens de Dados Informix a Servidores Federados” na página 47

Related reference:

- “db2set - DB2 Profile Registry Command” em *Command Reference*
- “ALTER SERVER statement” no *SQL Reference, Volume 2*

Capítulo 6. Configurando o Acesso às Origens de Dados do Oracle

Este capítulo explica como configurar seu servidor federado para acessar dados armazenados em origens de dados do Oracle.

Este capítulo lista as tarefas que precisam ser executadas, mostra exemplos das instruções SQL que precisam ser emitidas e fornece informações sobre ajuste e resolução de problemas que você pode utilizar ao fazer a configuração das origens de dados do Oracle.

Adicionando Origens de Dados Oracle a Servidores Federados

A configuração do servidor federado para acessar origens de dados Oracle envolve o fornecimento de informações ao servidor sobre as origens de dados Oracle e objetos que você deseja acessar.

Você pode configurar o acesso a origens de dados Oracle através do Centro de Controle do DB2, através do Centro de Comandos do DB2 ou através do processador da linha de comandos do DB2.

A vantagem de utilizar o Centro de Controle do DB2 é que não é necessário digitar cada instrução e comando. O Centro de Controle do DB2 fornece o método mais fácil de configurar rapidamente o acesso a origens de dados Oracle. No entanto, você não pode utilizar o Centro de Controle do DB2 para iniciar as seguintes tarefas de configuração:

- Configurar e testar o arquivo de configuração do cliente Oracle
- Testar a conexão com o servidor Oracle para validar a definição de servidor e mapeamentos de usuários
- Incluir ou eliminar opções de coluna

Pré-requisitos:

- Acesse o Centro de Comandos do DB2 ou o processador da linha de comandos do DB2.
- Um servidor federado e um banco de dados que são configurados para acessar origens de dados Oracle.
- O software do cliente Oracle que está instalado e configurado no servidor federado.

- A configuração apropriada de variáveis de ambiente do sistema, variáveis db2dj.ini e variáveis DB2 Profile Registry (db2set). As variáveis são: ORACLE_HOME, ORACLE_BASE, ORA_NLS e TNS_ADMIN.

Procedimento:

Para adicionar uma origem de dados Oracle a um servidor federado:

1. Configure e teste o arquivo de configuração do cliente Oracle.
2. Registre o wrapper.
3. Registre a definição de servidor.
4. Crie os mapeamentos do usuário.
5. Teste a conexão com o servidor Oracle.
6. Registre pseudônimos para tabelas e exibições do Oracle.

Related concepts:

- “Trilha Rápida para Configurar Origens de Dados” na página 1

Related tasks:

- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Configurando e Testando o Arquivo de Configuração do Cliente Oracle” na página 64
- “Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Oracle” na página 76

Configurando e Testando o Arquivo de Configuração do Cliente Oracle

Configurar e testar o arquivo de configuração do cliente Oracle fazem parte da maior tarefa de adicionar origens de dados Oracle a servidores federados.

O arquivo de configuração do cliente é utilizado para conectar-se a bancos de dados Oracle utilizando as bibliotecas de cliente que estão instaladas no servidor federado. Este arquivo especifica a localização de cada servidor do banco de dados Oracle e o tipo de conexão (protocolo) para o servidor do banco de dados. O nome padrão para o arquivo de configuração do cliente Oracle é tnsnames.ora.

Procedimento:

Para configurar e testar um arquivo de configuração do cliente Oracle:

1. Utilize o utilitário fornecido com o software do cliente Oracle.

Consulte a documentação da instalação do Oracle para obter mais informações sobre como utilizar esse utilitário. No arquivo tnsnames.ora, o SID (ou SERVICE_NAME) é o nome da instância do Oracle e HOST é o nome do host no qual o servidor Oracle está localizado.

O diretório no qual o arquivo tnsnames.ora é criado depende do sistema operacional em execução no servidor federado.

- Em sistemas operacionais UNIX, o caminho e nome padrão deste arquivo são \$ORACLE_HOME/network/admin .
 - Em sistemas operacionais Windows, o caminho e nome padrão deste arquivo são %ORACLE_HOME%\NETWORK\ADMIN.
2. Se desejar colocar o arquivo tnsnames.ora em um caminho diferente do caminho de pesquisa padrão, defina a variável de ambiente TNS_ADMIN para especificar a localização do arquivo.
 - a. Edite o arquivo db2dj.ini que está localizado no diretório sqlplus/cfg e defina a variável de ambiente TNS_ADMIN:

```
TNS_ADMIN=x:/path/
```
 - b. Emita os seguintes comandos para reciclar a instância do DB2 e para assegurar que a variável de ambiente esteja definida no programa:

```
db2stop
db2start
```
 3. Teste a conexão utilizando a ferramenta **sqlplus** do Oracle para assegurar que o software do cliente possa conectar-se ao servidor Oracle.

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper Oracle.

Related tasks:

- “Registrando o Wrapper Oracle” na página 65
- “Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Oracle” na página 76

Registrando o Wrapper Oracle

O registro do wrapper Oracle faz parte da maior tarefa de adicionar origens de dados Oracle a servidores federados.

Para especificar o wrapper que será utilizado para acessar origens de dados Oracle, utilize a instrução CREATE WRAPPER. Dois wrappers para Oracle, SQLNET e NET8, estão incluídos no DB2 Information Integrator.

Procedimento:

Para especificar o wrapper que deseja utilizar para acessar origens de dados Oracle, emita a instrução CREATE WRAPPER.

Por exemplo:

```
CREATE WRAPPER NET8
```

Para determinar qual nome de wrapper (SQLNET ou NET8) utilizar com a instrução CREATE WRAPPER, consulte a Referência Relacionada no final deste tópico.

O wrapper SQLNET utiliza chamadas de API OCI 7 (Oracle Call Interface). O wrapper NET8 utiliza chamadas de API OCI 8. Se o cliente Oracle 8 ou Oracle 9 estiver instalado, você obterá melhor desempenho e funcionalidade utilizando o wrapper NET8. Além disso, o wrapper NET8 possui suporte ao LOB. Como o OCI 7 não suporta tipos de dados LOB, o wrapper SQLNET não suporta tipos de dados LOB Oracle.

- O wrapper SQLNET mapeia tipos de dados LONG Oracle para tipos de dados LOB DB2 para UNIX e Windows.
- O wrapper NET8 não suporta tipos de dados LONG Oracle. Ele mapeia tipos de dados LOB Oracle para tipos de dados LOB DB2 para UNIX e Windows.

Recomendação: Utilize o nome do wrapper padrão (SQLNET ou NET8). Ao registrar o wrapper utilizando um dos nomes padrão, o servidor federado utilizará automaticamente o nome da biblioteca padrão associado a esse wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, será possível substituir o nome do wrapper padrão por um nome de sua escolha. Se você utilizar um nome diferente de um dos nomes padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.

Suponha que você tenha um servidor federado em execução no AIX e decida utilizar um nome de wrapper que não seja um dos nomes padrão. Exemplos de instruções CREATE WRAPPER para SQLNET e NET8 são:

```
CREATE WRAPPER mywrapper LIBRARY 'libdb2sqlnet.a'
```

```
CREATE WRAPPER mywrapper LIBRARY 'libdb2net8.a'
```

Consulte a Referência Relacionada no final deste tópico para obter uma lista de nomes de bibliotecas do wrapper Oracle.

A próxima tarefa nesta seqüência de tarefas é registrar definições de servidor para uma origem de dados Oracle.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Oracle” na página 68

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*
- “Nomes de Wrappers e de Bibliotecas Oracle” na página 67

Nomes de Wrappers e de Bibliotecas Oracle

Este tópico fornece os nomes de wrappers Oracle e nomes de bibliotecas do Oracle que você pode utilizar ao registrar um wrapper para acessar origens de dados Oracle.

A tabela a seguir lista os nomes de wrappers Oracle a serem utilizados, dependendo da versão do cliente Oracle e do sistema operacional que está sendo utilizado.

Tabela 13. Wrappers Oracle por versão de cliente e sistema operacional

| Cliente Oracle | Sistema operacional | Wrapper a ser utilizado |
|-----------------|---|-------------------------------|
| Oracle Versão 7 | AIX | SQLNET |
| | Windows NT e Windows 2000 | SQLNET |
| | Ambiente Operacional Solaris, HP-UX e Linux | não aplicável |
| Oracle Versão 8 | AIX | NET8 |
| | Windows NT ou Windows 2000 | NET8 (recomendável) ou SQLNET |
| | HP-UX, Linux e Solaris | NET8 |
| Oracle Versão 9 | AIX | NET8 |
| | Windows NT ou Windows 2000 | NET8 (recomendável) ou SQLNET |
| | HP-UX, Linux e Solaris | NET8 |

A tabela a seguir lista os nomes de bibliotecas do wrapper Oracle a serem utilizados, dependendo do sistema operacional do servidor federado.

Tabela 14. Nomes de bibliotecas do wrapper Oracle

| Sistema operacional no servidor federado | Nomes de bibliotecas do wrapper para SQLNET | Nomes de bibliotecas do wrapper para NET8 |
|--|---|---|
| AIX | libdb2sqlnet.a | libdb2net8.a |
| HP-UX | libdb2sqlnet.sl | libdb2net8.sl |
| Linux | libdb2sqlnet.so | libdb2net8.so |

Tabela 14. Nomes de bibliotecas do wrapper Oracle (continuação)

| Sistema operacional no servidor federado | Nomes de bibliotecas do wrapper para SQLNET | Nomes de bibliotecas do wrapper para NET8 |
|--|---|---|
| Ambiente Operacional Solaris | libdb2sqlnet.so | libdb2net8.so |
| Windows NT e Windows 2000 | db2sqlnet.dll | db2net8.dll |

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Oracle” na página 68

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando Definições de Servidor para uma Origem de Dados Oracle

O registro de definições de servidor para uma origem de dados Oracle faz parte da maior tarefa de adicionar origens de dados Oracle a servidores federados.

No banco de dados federado, você deve definir cada servidor Oracle que deseja acessar. Primeiro, você deve localizar o nome do nó da origem de dados Oracle e, em seguida, utilizar este nome do nó ao registrar o servidor.

Procedimento:

Para registrar uma definição de servidor para uma origem de dados Oracle:

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 o `node_name` esteja especificado como uma opção na instrução `SQL CREATE SERVER`, ele é obrigatório para origens de dados Oracle.

2. Emita a instrução `CREATE SERVER`.

Por exemplo:

```
CREATE SERVER server_name TYPE oracle VERSION 8.1.7 WRAPPER net8
  OPTIONS (NODE 'node_name')
```

Após a definição de servidor ter sido criada, utilize a instrução ALTER SERVER para incluir ou eliminar opções do servidor.

A próxima tarefa nesta seqüência de tarefas é criar mapeamentos de usuários para uma origem de dados Oracle.

Related tasks:

- “Criando Mapeamentos de Usuários para uma Origem de Dados Oracle” na página 71

Related reference:

- “ALTER SERVER statement” no *SQL Reference, Volume 2*
- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- Apêndice B, “Opções do Servidor para Sistemas Federados”, na página 391
- “Instrução CREATE SERVER - Exemplos para o Wrapper Oracle” na página 69

Instrução CREATE SERVER - Exemplos para o Wrapper Oracle

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE SERVER para registrar servidores para o wrapper Oracle. Este tópico inclui um exemplo completo, que mostra como registrar um servidor com parâmetros requeridos e um exemplo com opções de servidor adicionais.

Exemplo completo:

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

```
CREATE SERVER oraserver TYPE oracle VERSION 8.1.7 WRAPPER net8
  OPTIONS (NODE 'paris_node')
```

oraserver

Um nome atribuído ao servidor do banco de dados Oracle. Esse nome deve ser exclusivo. Não são permitidos nomes de servidores duplicados.

TYPE *oracle*

Especifica o tipo de servidor de origem de dados para o qual você está configurando acesso. O parâmetro type para os wrappers SQLNET e NET8 deve ser *oracle*.

VERSION 8.1.7

A versão do servidor do banco de dados Oracle que você deseja acessar. As versões do Oracle suportadas são 7.3.4, 8.x e 9.x.

WRAPPER *net8*

O nome especificado na instrução CREATE WRAPPER.

NODE 'paris_node'

O nome do nó no qual o servidor do banco de dados Oracle reside. Obtenha o nome do nó do arquivo tnsnames.ora.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dados Oracle.

Exemplo de opção do servidor:

Ao criar a definição de servidor você pode especificar opções de servidores adicionais na instrução CREATE SERVER. Estas opções de servidor incluem opções gerais de servidor e opções de servidor específicas do Oracle.

O DB2 assume que todas as colunas VARCHAR do Oracle contêm espaços em branco finais. Se tiver certeza de que todas as colunas VARCHAR no banco de dados Oracle não contêm espaços em branco finais, poderá definir uma opção de servidor para especificar que a origem de dados utiliza uma semântica de comparação VARCHAR não preenchida por espaços em branco.

O exemplo a seguir mostra uma definição de servidor Oracle com esta opção de servidor:

```
CREATE SERVER oraserver TYPE oracle VERSION 8.1.7 WRAPPER net8
  OPTIONS (NODE 'paris_node', VARCHAR_NO_TRAILING_BLANKS 'Y')
```

Utilize a opção de servidor VARCHAR_NO_TRAILING_BLANKS quando nenhuma das colunas contiver espaços em branco finais. Se apenas algumas das colunas VARCHAR contiverem espaços em branco finais, será possível definir uma opção nessas colunas específicas com a instrução ALTER NICKNAME.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Oracle” na página 68

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Criando Mapeamentos de Usuários para uma Origem de Dados Oracle

A criação de mapeamentos de usuários para uma origem de dados Oracle faz parte da maior tarefa de adicionar origens de dados Oracle a servidores federados.

Ao tentar acessar um servidor Oracle, o servidor federado estabelece uma conexão com a origem de dados utilizando um ID do usuário e senha válidos para essa origem de dados. É necessário definir uma associação (mapeamento do usuário) entre cada ID do usuário do servidor federado e o ID do usuário e senha da origem de dados correspondente. Crie um mapeamento de usuário para cada ID do usuário que acessará o sistema federado para enviar pedidos distribuídos para a origem de dados Oracle.

Procedimento:

Para mapear um ID do usuário local para o ID do usuário e senha do servidor Oracle, emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR userid SERVER oraserver
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

A próxima tarefa nesta seqüência de tarefas é testar a conexão com o servidor Oracle.

Related tasks:

- “Testando a Conexão com o Servidor Oracle” na página 73

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Instrução CREATE USER MAPPING - Exemplos para o Wrapper Oracle” na página 71

Instrução CREATE USER MAPPING - Exemplos para o Wrapper Oracle

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE USER MAPPING para mapear um ID do usuário do servidor federado para um ID do usuário e senha do servidor Oracle. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o registro especial do DB2 USER com a instrução CREATE USER MAPPING.

Exemplo completo:

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

```
CREATE USER MAPPING FOR robert SERVER oraserver
    OPTIONS (REMOTE_AUTHID 'rob', REMOTE_PASSWORD 'then4now')
```

robert Especifica o ID do usuário local que você está mapeamento para um ID do usuário definido em um servidor Oracle.

SERVER *oraserver*

Especifica o nome do servidor Oracle definido na instrução CREATE SERVER.

REMOTE_AUTHID '*rob*'

Especifica o ID do usuário no servidor do banco de dados Oracle para o qual você está mapeando *robert*. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_ID como 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD '*then4now*'

Especifica a senha associada a '*rob*'. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_PW como 'U' ou 'L' na instrução CREATE SERVER.

Exemplo especial de registro:

O exemplo a seguir mostra uma instrução CREATE USER MAPPING que inclui o registro especial USER:

```
CREATE USER MAPPING FOR USER SERVER oraserver
    OPTIONS (REMOTE_AUTHID 'rob', REMOTE_PASSWORD 'then4now')
```

Você pode utilizar o registro especial USER 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 REMOTE_AUTHID do usuário.

Restrição: O ID do usuário na origem de dados Oracle deve ter sido criado utilizando o comando **create user** do Oracle com a cláusula 'identified by' em vez da cláusula 'identified externally'.

Related tasks:

- “Criando Mapeamentos de Usuários para uma Origem de Dados Oracle” na página 71

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Testando a Conexão com o Servidor Oracle

O teste da conexão com o servidor Oracle faz parte da maior tarefa de adicionar origens de dados Oracle a servidores federados.

Você pode testar a conexão com o servidor Oracle utilizando a definição de servidor e mapeamentos de usuários definidos.

Procedimento:

Para testar a conexão:

1. Abra uma sessão de passagem para emitir uma instrução SQL SELECT nas tabelas do sistema Oracle.

Por exemplo:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM sys.all_tables
SET PASSTHRU RESET
```

Se a instrução SQL SELECT retornar uma contagem, a definição de servidor e o mapeamento do usuário estão configurados adequadamente.

2. Se a instrução SQL SELECT retornar um erro, é possível que seja necessário:
 - Verificar o servidor Oracle para certificar-se de que ele esteja configurado para conexões de entrada.
 - Verificar o mapeamento de usuário para certificar-se de que as definições para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com o servidor Oracle. Alterar o mapeamento de usuário ou criar outro mapeamento, conforme necessário.
 - Verificar o software de cliente Oracle no servidor federado DB2 para certificar-se de que ele esteja instalado e configurado corretamente para conectar-se ao servidor Oracle.
 - Verificar as definições das variáveis federadas do DB2 para verificar se estão corretas para o servidor Oracle. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis db2dj.ini e a variável DB2 Profile Registry (db2set).
 - Verifique sua definição de servidor. Se necessário, elimine-a e crie-a novamente.

A próxima etapa nesta sequência de tarefas é registrar pseudônimos para tabelas e exibições do Oracle.

Related concepts:

- “Definições do Servidor e Opções do Servidor” na página 10

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Registrando Pseudônimos para Tabelas e Exibições do Oracle” na página 74

Related reference:

- “ALTER USER MAPPING statement” no *SQL Reference, Volume 2*

Registrando Pseudônimos para Tabelas e Exibições do Oracle

O registro de pseudônimos para tabelas de exibições do Oracle faz parte da maior tarefa de adicionar origens de dados Oracle a servidores federados.

Para cada servidor Oracle definido, registre um pseudônimo para cada tabela ou exibição que deseja acessar. Utilize estes pseudônimos, em vez dos nomes de objetos de origem de dados, quando consultar os servidores Oracle.

O banco de dados federado depende das estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento da consulta. Essas estatísticas são coletadas ao registrar um pseudônimo para um objeto da origem de dados utilizando a instrução CREATE NICKNAME. O banco de dados federado verifica a presença do objeto na origem de dados e tenta coletar os dados estatísticos existentes da origem de dados. As informações úteis ao otimizador são lidas a partir dos catálogos de origem de dados e colocadas no catálogo global no servidor federado. Como algumas informações do catálogo de origem de dados, ou todas elas, podem ser utilizadas pelo otimizador, atualize as estatísticas (utilizando o comando de origem de dados equivalente ao comando RUNSTATS do DB2) na origem de dados antes de registrar um pseudônimo.

Procedimento:

Para registrar um pseudônimo, emita a instrução CREATE NICKNAME.

Por exemplo:

```
CREATE NICKNAME oracle_name FOR oraserver."remote_schema"."remote.table"
```

Os pseudônimos podem ter até 128 caracteres.

Repita esta etapa para cada tabela ou exibição do Oracle para a qual você deseja criar um pseudônimo.

Quando você cria o pseudônimo, o DB2 utilizará a conexão para consultar o catálogo de origens de dados. Esta consulta testa a conexão com a origem de dados utilizando o pseudônimo. Se a conexão não funcionar, você receberá uma mensagem de erro.

Related reference:

- “RUNSTATS Command” na publicação *Command Reference*
- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper Oracle” na página 75

Instrução CREATE NICKNAME - Exemplos para o Wrapper Oracle

Este tópico fornece um exemplo que mostra como utilizar a instrução CREATE NICKNAME para registrar um pseudônimo para uma tabela ou exibição do Oracle que você deseja acessar.

Este exemplo mostra como especificar um objeto remoto para o servidor Oracle com o qual o pseudônimo é atribuído:

```
CREATE NICKNAME PARISINV FOR oraserver."france"."inventory"
```

PARISINV

É um pseudônimo exclusivo utilizado para identificar a tabela ou exibição do Oracle.

Nota: o pseudônimo é um nome com duas partes — o esquema e o pseudônimo. Se omitir o esquema ao registrar o pseudônimo, o esquema do pseudônimo será o ID de autorização do usuário que está registrando o pseudônimo.

oraserver."france"."inventory"

Um identificador de três partes do objeto remoto:

- *oraserver* é o nome atribuído ao servidor do banco de dados Oracle na instrução CREATE SERVER.
- *france* é o nome do esquema remoto ao qual pertence a tabela ou exibição.
- *inventory* é o nome da tabela ou exibição remota que você deseja acessar.

O servidor federado dobra nomes de esquemas e tabelas do Oracle para maiúsculas, a menos que você coloque os nomes entre aspas.

Related tasks:

- “Registrando Pseudônimos para Tabelas e Exibições do Oracle” na página 74

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*

Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Oracle

Depois de fazer a configuração para origens de dados Oracle, você poderá modificar a configuração para aprimorar o desempenho. Por exemplo, talvez você queira definir a variável de registro de perfil DB2_DJ_COMM para aprimorar o desempenho quando a origem de dados Oracle for acessada.

Aprimorando o Desempenho Definindo a Variável DB2_DJ_COMM (UNIX)

Se perceber que demora muito tempo para acessar o servidor Oracle, você poderá aprimorar o desempenho definindo a variável DB2_DJ_COMM. Quando definir DB2_DJ_COMM, o servidor federado carregará o wrapper durante a inicialização em vez de quando você tentar acessar a origem de dados.

Procedimento:

Para definir a variável DB2_DJ_COMM:

1. Defina a variável DB2_DJ_COMM para o wrapper da biblioteca correspondente ao wrapper especificado. Suponha que o servidor federado esteja executando o AIX e o wrapper que está sendo utilizado seja NET8. O comando para definir a variável DB2_DJ_COMM é:

```
db2set DB2_DJ_COMM= 'libdb2net8.a'
```

A tabela a seguir lista os nomes de bibliotecas do Oracle válidas.

Tabela 15. Nomes de bibliotecas do wrapper Oracle

| Sistema operacional no servidor federado | Nomes de bibliotecas do wrapper SQLNET | Nomes de bibliotecas do wrapper NET8 |
|--|--|--------------------------------------|
| AIX | libdb2sqlnet.a | libdb2net8.a |
| HP-UX | libdb2sqlnet.sl | libdb2net8.sl |
| Linux | libdb2sqlnet.so | libdb2net8.so |
| Ambiente Operacional Solaris | libdb2sqlnet.so | libdb2net8.so |

2. Emita os comandos a seguir, para reciclar a instância do DB2:

```
db2stop  
db2start
```

Ao reciclar a instância do DB2, assegure que ela aceite alterações de variáveis feitas.

Problemas de Conectividade

Para cada HOST na seção DESCRIPTION do arquivo tnsnames.ora, talvez seja necessário atualizar o arquivo hosts. A condição para você atualizar este arquivo depende de como o TCP/IP está configurado em sua rede. Parte da rede deve converter o nome do host remoto que está especificado na seção DESCRIPTION no arquivo tnsnames.ora em um endereço.

Se sua rede tiver um servidor denominado que reconheça o nome do host, não será necessário atualizar o arquivo hosts do TCP/IP. De outra maneira, você precisa de uma entrada para o host remoto. Consulte o administrador da rede para determinar como sua rede está configurada. Se precisar atualizar o arquivo hosts, a localização do arquivo dependerá do sistema operacional do servidor federado:

Em servidores federados UNIX

Atualize o arquivo /etc/hosts.

Em servidores federados Windows

Atualize o arquivo x:\winnt\system32\drivers\etc\hosts.

Related tasks:

- “Adicionando Origens de Dados Oracle a Servidores Federados” na página 63

Related reference:

- “db2set - DB2 Profile Registry Command” em *Command Reference*

Capítulo 7. Configurando o Acesso às Origens de Dados do Sybase

Este capítulo explica como configurar seu servidor federado para acessar dados armazenados em origens de dados do Sybase.

Este capítulo lista as tarefas que precisam ser executadas, mostra exemplos das instruções SQL que precisam ser emitidas e fornece informações sobre ajuste e resolução de problemas que você pode utilizar ao fazer a configuração das origens de dados do Sybase.

Adicionando Origens de Dados Sybase a Servidores Federados

A configuração do servidor federado para acessar origens de dados Sybase envolve o fornecimento de informações ao servidor sobre as origens de dados Sybase e objetos que você deseja acessar.

Você pode configurar o acesso a origens de dados Sybase através do Centro de Controle do DB2, através do Centro de Comandos do DB2 ou através do processador da linha de comandos do DB2.

A vantagem de utilizar o Centro de Controle do DB2 é que não é necessário digitar cada instrução e comando. O Centro de Controle do DB2 fornece o método mais fácil de configurar rapidamente o acesso a origens de dados Sybase. No entanto, você não pode utilizar o Centro de Controle do DB2 para iniciar as seguintes tarefas de configuração:

- Configurar e testar o arquivo de configuração do cliente Sybase
- Testar a conexão com o servidor Sybase para validar a definição de servidor e mapeamentos de usuários
- Incluir ou eliminar opções de coluna

Pré-requisitos:

- Acesse o Centro de Comandos do DB2 ou o processador da linha de comandos do DB2.
- Um servidor federado e um banco de dados que são configurados para acessar origens de dados Sybase.
- O software do cliente Sybase que está instalado e configurado no servidor federado.

- A configuração apropriada de variáveis de ambiente do sistema, variáveis db2dj.ini e variáveis DB2 Profile Registry (db2set). As variáveis são: SYBASE e SYBASE_OCS.

Restrição:

O wrapper Sybase Open Client DB-Library chamado DBLIB é um wrapper somente leitura e não suporta operações INSERT, UPDATE ou DELETE.

Procedimento:

Para adicionar uma origem de dados Sybase a um servidor federado:

1. Configure e teste o arquivo de configuração do cliente Sybase.
2. Registre o wrapper.
3. Registre a definição de servidor.
4. Crie os mapeamentos do usuário.
5. Teste a conexão com o servidor Sybase.
6. Registre pseudônimos para tabelas e exibições do Sybase.

Related concepts:

- “Trilha Rápida para Configurar Origens de Dados” na página 1

Related tasks:

- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Configurando e Testando o Arquivo de Configuração do Cliente Sybase” na página 80

Configurando e Testando o Arquivo de Configuração do Cliente Sybase

Configurar e testar o arquivo de configuração do cliente Sybase fazem parte da maior tarefa de adicionar origens de dados Sybase a servidores federados.

O arquivo de configuração do cliente é utilizado para conectar-se ao Sybase utilizando bibliotecas do Sybase Open Client que estão instaladas no servidor federado. Este arquivo especifica a localização de cada instância do Sybase SQL Server e do Adaptive Server Enterprise e tipo de conexão (protocolo) para o servidor do banco de dados.

Você deve configurar um arquivo de configuração em cada instância no servidor federado DB2 que será utilizado para conectar-se ao Sybase. As

etapas que devem ser utilizadas para configurar e testar este arquivo dependem do sistema operacional no qual você está executando seu servidor federado.

Procedimento:

Para configurar e testar um arquivo de configuração do cliente:

Em sistemas operacionais UNIX:

1. Configure o arquivo de configuração do cliente utilizando o utilitário que é fornecido com o software Sybase Open Client. Este arquivo é criado no diretório \$SYBASE/interfaces. Consulte a documentação do Sybase para obter mais informações sobre como utilizar este utilitário.
2. Torne o arquivo interfaces acessível à instância do servidor federado DB2 utilizando um dos seguintes métodos:
 - Copie este arquivo para o diretório \$HOME/sqllib da instância federada do DB2.
 - Utilize o comando **ln** para criar um link do subdiretório /sqllib para o arquivo interfaces no diretório \$HOME/sqllib da instância. Por exemplo:

```
ln -s -f /home/sybase/interfaces /home/db2djinst1/sqllib
```
 - Utilize a opção de servidor IFILE para especificar o caminho completo para o arquivo interfaces do Sybase.
3. Teste a conexão para assegurar que o software Sybase Open Client possa conectar-se ao servidor Sybase. Utilize um utilitário de consulta do Sybase apropriado, tal como, **isql**.

Em sistemas operacionais Windows:

1. Configure o arquivo de configuração do cliente utilizando o utilitário que é fornecido com o software Sybase Open Client. Este arquivo é criado no diretório %SYBASE%\ini\sql.ini. Consulte a documentação do Sybase para obter mais informações sobre como utilizar este utilitário.
2. Torne este arquivo sql.ini acessível à instância do servidor federado DB2, copiando este arquivo para o diretório c:\Arquivos de Programas\IBM\SQLLIB da instância federada do DB2.

Como o DB2 Information Integrator utiliza interfaces como o nome padrão para o arquivo de configuração do cliente Sybase, renomeie o arquivo sql.ini do Windows no diretório c:\Arquivos de Programas\IBM\SQLLIB para interfaces.

Obrigatório: Se você não renomear o arquivo sql.ini para interfaces, deverá utilizar a opção de servidor IFILE quando criar a definição de servidor.

3. Teste a conexão para assegurar que o software Sybase Open Client possa conectar-se ao servidor Sybase. Utilize um utilitário de consulta do Sybase apropriado, tal como, **isql**.

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper Sybase.

Related tasks:

- “Registrando o Wrapper Sybase” na página 82

Registrando o Wrapper Sybase

O registro do wrapper Sybase faz parte da maior tarefa de adicionar origens de dados Sybase a servidores federados.

Para especificar o wrapper que será utilizado para acessar origens de dados Sybase, utilize a instrução CREATE WRAPPER. Dois wrappers para Sybase, o wrapper Open Client Client-Library chamado CTLIB e o wrapper Open Client DB-Library chamado DBLIB, estão incluídos no DB2 Information Integrator.

Procedimento:

Para especificar o wrapper que deseja utilizar para acessar origens de dados Sybase, emita a instrução CREATE WRAPPER.

Por exemplo:

```
CREATE WRAPPER CTLIB
```

Você pode utilizar o wrapper CTLIB ou DBLIB, independentemente do sistema operacional que está em execução em seu servidor federado.

Recomendação: Utilize um dos nomes de wrappers padrão (CTLIB ou DBLIB). Ao registrar o wrapper utilizando um dos nomes padrão, o servidor federado utiliza automaticamente o nome da biblioteca padrão que está associada a esse wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, você poderá substituir o nome do wrapper padrão por um nome de sua escolha. Se você utilizar um nome diferente de um dos nomes padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER. Suponha que você tenha um servidor federado em execução no AIX e decida utilizar um nome de wrapper que não seja um dos nomes padrão. Exemplos das instruções CREATE WRAPPER para CTLIB e DBLIB são:

```
CREATE WRAPPER mywrapper LIBRARY 'libdb2ctlib.a'
```

```
CREATE WRAPPER mywrapper LIBRARY 'libdb2dblib.a'
```

Os nomes de bibliotecas de wrapper para o Sybase são:

Tabela 16. Nomes de bibliotecas do wrapper Sybase

| Sistema operacional no servidor federado | Nomes de bibliotecas do wrapper CTLIB | Nomes de bibliotecas do wrapper DBLIB |
|--|---------------------------------------|---------------------------------------|
| AIX | libdb2ctlib.a | libdb2dblib.a |
| HP-UX | libdb2ctlib.sl | libdb2dblib.sl |
| Linux | libdb2ctlib.so | libdb2dblib.so |
| Ambiente Operacional Solaris | libdb2ctlib.so | libdb2dblib.so |
| Windows NT e Windows 2000 | db2ctlib.dll | db2dblib.dll |

A próxima tarefa nesta seqüência de tarefas é registrar as definições de servidor para uma origem de dados Sybase.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Sybase” na página 83

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando Definições de Servidor para uma Origem de Dados Sybase

O registro de definições de servidor para uma origem de dados Sybase faz parte da maior tarefa de adicionar origens de dados Sybase a servidores federados.

No banco de dados federado, você deve definir cada servidor Sybase que deseja acessar. Primeiro, você deve localizar o nome do nó da origem de dados Sybase e, em seguida, utilizar este nome do nó ao registrar o servidor.

Procedimento:

Para registrar uma definição de servidor para uma origem de dados Sybase:

1. Localize o nome do nó no arquivo interfaces do Sybase.

Arquivo interfaces de exemplo em sistemas operacionais UNIX:

```
sybase119
query tcp ether anaconda 4100
```

Arquivo interfaces de exemplo em sistemas operacionais Windows NT ou Windows 2000:

```
[sybase119]
query=TCP,anaconda,4100
```

Nestes exemplos, o nome do nó é sybase119. O nome do nó é seguido pelo tipo de conexão (TCP/IP) e pelo nome do host (anaconda).

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dados Sybase.

2. Emita a instrução CREATE SERVER.

Por exemplo:

```
CREATE SERVER server_name TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
  OPTIONS (NODE 'sybnode', DBNAME 'sybdb')
```

Após a definição de servidor ter sido criada, utilize a instrução ALTER SERVER para incluir ou eliminar opções do servidor.

Importante: Se você não renomeou o arquivo sql.ini para interfaces quando configurou o arquivo de configuração do cliente Sybase, deverá utilizar a opção de servidor IFILE quando registrar a definição de servidor.

A próxima tarefa nesta seqüência de tarefas é criar um mapeamento de usuário para uma origem de dados Sybase.

Related tasks:

- “Criando um Mapeamento de Usuário para uma Origem de Dados Sybase” na página 87

Related reference:

- “ALTER SERVER statement” no *SQL Reference, Volume 2*
- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- Apêndice B, “Opções do Servidor para Sistemas Federados”, na página 391
- “Instrução CREATE SERVER - Exemplos para o Wrapper Sybase” na página 84

Instrução CREATE SERVER - Exemplos para o Wrapper Sybase

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE SERVER para registrar servidores para o wrapper Sybase. Este tópico inclui um exemplo completo, que mostra como registrar um servidor com parâmetros requeridos e um exemplo com opções de servidor adicionais.

Exemplo completo:

O exemplo a seguir mostra como registrar uma definição de servidor para um wrapper Sybase, emitindo a instrução CREATE SERVER:

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB  
OPTIONS (NODE 'sybnode', DBNAME 'sybdb')
```

SYBSERVER

Um nome atribuído ao servidor Sybase. Esse nome deve ser exclusivo. Não são permitidos nomes de servidores duplicados.

TYPE *SYBASE*

Especifica Sybase como o tipo de origem de dados para o qual você está configurando o acesso. O parâmetro TYPE para os wrappers CTLIB e DBLIB deve ser *SYBASE*.

VERSION *12.0*

A versão de software do servidor do banco de dados Sybase que você deseja acessar. As versões suportadas são 11, 11.5, 11.9, 12 e 12.5.

WRAPPER *CTLIB*

O nome do wrapper especificado na instrução CREATE WRAPPER.

NODE *'sybnode'*

O nome do nó no qual *SYBSERVER* reside. Obtenha o nome do nó a partir do arquivo interfaces. Este valor faz distinção entre letras maiúsculas e minúsculas.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dados Sybase.

DBNAME *'sybdb'*

O nome do banco de dados Sybase que você deseja acessar. Obtenha este nome do servidor Sybase. Este valor faz distinção entre letras 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 Sybase.

Importante: Se você não renomeou o arquivo sql.ini para interfaces quando configurou o arquivo de configuração do cliente Sybase, deverá utilizar a opção de servidor IFILE ao registrar a definição de servidor.

Exemplos de opções de servidor:

Ao registrar o servidor, você pode especificar opções de servidor adicionais na instrução CREATE SERVER. Estas opções de servidor incluem opções gerais de servidor e opções de servidor específicas do Sybase.

O exemplo a seguir mostra como utilizar a opção de servidor TIMEOUT ao registrar um servidor com o wrapper CTLIB em um sistema operacional UNIX:

```
CREATE SERVER SYBSERVER TYPE SYBASE
VERSION 12.0 WRAPPER CTLIB
      OPTIONS (NODE 'sybnode', DBNAME 'sybdb',
              TIMEOUT '60', LOGIN_TIMEOUT '60', PACKET_SIZE '1024',
              IFILE '/home/sybase/interfaces')
```

O valor de tempo limite é o número de segundos que o wrapper aguarda uma resposta do servidor Sybase. Utilize a opção TIMEOUT para evitar congelamentos em transações.

O exemplo a seguir mostra como utilizar a opção de servidor IFILE ao registrar um servidor em um sistema operacional Windows:

```
CREATE SERVER SYBSERVER TYPE SYBASE
VERSION 12.0 WRAPPER CTLIB
      OPTIONS (NODE 'sybnode', DBNAME 'sybdb',
              IFILE 'C:\Sybase\ini\sql.ini')
```

As opções adicionais de servidor específicas do Sybase são:

IFILE

Especifica o caminho e nome completo do arquivo interfaces do Sybase Open Client.

Utilize esta opção de servidor se você não copiou ou vinculou o arquivo sql.ini como \$SQLLIB\interfaces (em sistemas UNIX) ou como %SQLLIB%/interfaces (em sistemas operacionais Windows).

IGNORE_UDT

Especifica se o servidor federado determina o tipo interno que suporta um UDT sem um tipo sólido.

LOGIN_TIMEOUT

Especifica o período de tempo, em segundos, que o DB2 Universal Database aguarda uma resposta de login ao fazer uma tentativa de conexão. O comportamento padrão é aguardar indefinidamente uma resposta do servidor Sybase.

PACKET_SIZE

Determina o tamanho do pacote que o Client-Library utiliza ao enviar pacotes TDS (Tabular Data Stream). Se um aplicativo precisar enviar ou receber grandes quantidades de texto, imagem ou dados em massa, um tamanho de pacote maior poderá aprimorar a eficiência.

TIMEOUT

Especifica o período de tempo, em segundos, que o DB2 Universal Database aguarda uma resposta do servidor para um comando. O comportamento padrão é aguardar indefinidamente uma resposta do

servidor Sybase. O Sybase Open Client utiliza limites de tempo limite para interromper consultas e respostas que são executadas por um longo período de tempo.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Sybase” na página 83

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Criando um Mapeamento de Usuário para uma Origem de Dados Sybase

A criação de um mapeamento de usuário para uma origem de dados Sybase faz parte da maior tarefa de adicionar origens de dados Sybase a servidores federados.

Ao tentar acessar um servidor Sybase, o servidor federado estabelece uma conexão com a origem de dados utilizando um ID do usuário e senha válidos para origem de dados. É necessário definir uma associação (mapeamento do usuário) entre cada ID do usuário do servidor federado e o ID do usuário e senha da origem de dados correspondente. Crie um mapeamento de usuário para cada ID do usuário que acessará o sistema federado para enviar pedidos distribuídos para a origem de dados Sybase.

Procedimento:

Para mapear um ID do usuário local para o ID do usuário e senha do servidor Sybase, emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR userid SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

A próxima tarefa nesta seqüência de tarefas é testar a conexão com o servidor Sybase.

Related tasks:

- “Testando a Conexão com o Servidor Sybase” na página 89

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Instrução CREATE USER MAPPING - Exemplos para o Wrapper Sybase” na página 88

Instrução CREATE USER MAPPING - Exemplos para o Wrapper Sybase

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE USER MAPPING para mapear um ID do usuário do servidor federado para um ID do usuário e senha do Sybase. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o registro especial do DB2 USER com a instrução CREATE USER MAPPING.

Exemplo completo:

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

```
CREATE USER MAPPING FOR maria SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'mary', REMOTE_PASSWORD 'day2night')
```

maria Especifica o ID do usuário local que você está mapeando para um ID do usuário definido no servidor Sybase.

SERVER SYBSERVER

Especifica o nome do servidor Sybase definido na instrução CREATE SERVER.

REMOTE_AUTHID '*mary*'

Especifica o ID do usuário no servidor Sybase para o qual você está mapeando *maria*. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_ID como 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD '*day2night*'

Especifica a senha associada a '*mary*'. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_PW como 'U' ou 'L' na instrução CREATE SERVER.

Exemplo especial de registro:

O exemplo a seguir mostra uma instrução CREATE USER MAPPING que inclui o registro especial USER:

```
CREATE USER MAPPING FOR USER SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'mary', REMOTE_PASSWORD 'day2night')
```

Você pode utilizar o registro especial USER 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 REMOTE_AUTHID do usuário.

Related tasks:

- “Criando um Mapeamento de Usuário para uma Origem de Dados Sybase” na página 87

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Testando a Conexão com o Servidor Sybase

O teste da conexão com o servidor Sybase faz parte da maior tarefa de adicionar origens de dados Sybase a servidores federados.

Você pode testar a conexão com o servidor Sybase utilizando a definição de servidor e mapeamentos de usuários definidos.

Procedimento:

Para testar a conexão:

1. Abra uma sessão de passagem para emitir uma instrução SQL SELECT nas tabelas do sistema Sybase.

Por exemplo:

```
SET PASSTHRU local_server_name
SELECT count(*) FROM dbo.sysobjects
SET PASSTHRU RESET
```

Em que *local_server_name* é o nome utilizado para registrar o servidor remoto no catálogo do banco de dados federado. Se a instrução SQL SELECT retornar uma contagem, a definição de servidor e o mapeamento do usuário estão configurados adequadamente.

2. Se a instrução SQL SELECT retornar um erro, é possível que seja necessário:
 - Verificar o servidor Sybase para certificar-se de que ele esteja configurado para conexões de entrada.
 - Verificar o mapeamento de usuário para certificar-se de que as definições para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com o servidor Sybase. Alterar o mapeamento de usuário ou criar outro mapeamento, conforme necessário.
 - Verificar o software de cliente Sybase no servidor federado DB2 para certificar-se de que ele esteja instalado e configurado corretamente para conectar-se ao servidor Sybase.
 - Verificar as definições das variáveis federadas do DB2 para verificar se estão corretas para o servidor Sybase. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis db2dj.ini e a variável DB2 Profile Registry (db2set).
 - Verifique sua definição de servidor. Se necessário, elimine-a e crie-a novamente.

A próxima etapa nesta seqüência de tarefas é registrar pseudônimos para tabelas e exibições.

Related concepts:

- “Definições do Servidor e Opções do Servidor” na página 10

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Registrando Pseudônimos para Tabelas e Exibições do Sybase” na página 90

Related reference:

- “ALTER USER MAPPING statement” no *SQL Reference, Volume 2*

Registrando Pseudônimos para Tabelas e Exibições do Sybase

O registro de pseudônimos para tabelas de exibições do Sybase faz parte da maior tarefa de adicionar origens de dados Sybase a servidores federados.

Para cada servidor Sybase definido, registre um pseudônimo para cada tabela ou exibição que deseja acessar. Utilize estes pseudônimos, em vez dos nomes de objetos de origem de dados, quando consultar os servidores Sybase.

O banco de dados federado depende das estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento da consulta. Essas estatísticas são coletadas ao registrar um pseudônimo para um objeto da origem de dados utilizando a instrução CREATE NICKNAME. O banco de dados federado verifica a presença do objeto na origem de dados e tenta coletar os dados estatísticos existentes da origem de dados. As informações úteis ao otimizador são lidas a partir dos catálogos de origem de dados e colocadas no catálogo global no servidor federado. Como algumas informações do catálogo de origem de dados, ou todas elas, podem ser utilizadas pelo otimizador, atualize as estatísticas (utilizando o comando de origem de dados equivalente ao comando RUNSTATS do DB2) na origem de dados antes de registrar um pseudônimo.

Procedimento:

Para registrar um pseudônimo, emita a instrução CREATE NICKNAME.

Por exemplo:

```
CREATE NICKNAME sybase_name FOR SYBSERVER."remote_schema"."remote.table"
```

Os pseudônimos podem ter até 128 caracteres.

Repita esta etapa para cada tabela ou exibição do Sybase para a qual você deseja criar um pseudônimo.

Quando você cria o pseudônimo, o DB2 utilizará a conexão para consultar o catálogo de origens de dados. Esta consulta testa a conexão com a origem de dados utilizando o pseudônimo. Se a conexão não funcionar, você receberá uma mensagem de erro.

Related reference:

- “RUNSTATS Command” na publicação *Command Reference*
- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper Sybase” na página 91

Instrução CREATE NICKNAME - Exemplos para o Wrapper Sybase

Este tópico fornece um exemplo que mostra como utilizar a instrução CREATE NICKNAME para registrar um pseudônimo para uma tabela ou exibição do Sybase que você deseja acessar.

Este exemplo mostra como especificar um objeto remoto para o servidor Sybase com o qual o pseudônimo é atribuído:

```
CREATE NICKNAME SYBSALES FOR SYBSERVER."salesdata"."europa"
```

SYBSALES

É um pseudônimo único para a tabela ou a exibição Sybase.

O pseudônimo é um nome com duas partes — o esquema e o pseudônimo. Se omitir o esquema quando criar o pseudônimo, ele será o ID de autenticação do usuário que criou o pseudônimo.

*SYBSERVER."*salesdata"*."*europa"

É um identificador de três partes do objeto remoto.

- *SYBSERVER* é o nome atribuído ao servidor do banco de dados Sybase na instrução CREATE SERVER.
- *salesdata* é o nome do esquema remoto ao qual a tabela ou exibição pertence.
- *europa* é o nome da tabela ou exibição remota que você deseja acessar.

O servidor federado dobra nomes de esquemas e tabelas do Sybase para maiúsculas, a menos que você coloque os nomes entre aspas.

Related tasks:

- “Registrando Pseudônimos para Tabelas e Exibições do Sybase” na página 90

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*

Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Sybase

Depois de fazer a configuração para origens de dados Sybase, talvez você queira modificar a configuração para aprimorar o desempenho. Por exemplo, talvez você queira definir a variável de ambiente DB2_DJ_COMM para aprimorar o desempenho quando a origem de dados Sybase for acessada.

Aprimorando o Desempenho Definindo a Variável de Ambiente DB2_DJ_COMM (UNIX)

Se perceber que demora muito tempo para acessar o servidor Sybase, você poderá aprimorar o desempenho, definindo a variável de ambiente DB2_DJ_COMM. Ao definir a variável de ambiente DB2_DJ_COMM, o servidor federado carrega o wrapper durante a inicialização em vez de quando você tenta acessar a origem de dados.

Procedimento:

Para definir a variável de ambiente DB2_DJ_COMM:

1. Defina a variável de ambiente DB2_DJ_COMM para a biblioteca do wrapper que corresponde ao wrapper especificado. Suponha que o servidor federado esteja executando o AIX e o wrapper que está sendo utilizado seja CTLIB. O comando para definir a variável de ambiente DB2_DJ_COMM é:

```
db2set DB2_DJ_COMM= 'libdb2ctlib.a'
```

Consulte a tabela a seguir para o nome de biblioteca correto.

Tabela 17. Nomes de bibliotecas do wrapper Sybase

| Sistema operacional no servidor federado | Nomes de bibliotecas do wrapper CTLIB | Nomes de bibliotecas do wrapper DBLIB |
|--|---------------------------------------|---------------------------------------|
| AIX | libdb2ctlib.a | libdb2dblib.a |
| HP-UX | libdb2ctlib.sl | libdb2dblib.sl |

Tabela 17. Nomes de bibliotecas do wrapper Sybase (continuação)

| Sistema operacional no servidor federado | Nomes de bibliotecas do wrapper CTLIB | Nomes de bibliotecas do wrapper DBLIB |
|--|---------------------------------------|---------------------------------------|
| Linux | libdb2ctlib.so | libdb2dblib.so |
| Ambiente Operacional Solaris | libdb2ctlib.so | libdb2dblib.so |

2. Emita os comandos a seguir, para reciclar a instância do DB2:

```
db2stop  
db2start
```

Ao reciclar a instância do DB2, assegure que ela aceite alterações de variáveis feitas.

Utilizando CTLIB em Vez de DBLIB

CT-Library suporta preparação e execução de instruções dinâmicas. Isto permite que aplicativos CT-Library preparem uma instrução de uma etapa e execute-a muitas vezes com diferentes entradas. A preparação de uma instrução de uma etapa elimina a necessidade de recompilar a instrução para cada alteração de parâmetro de entrada. Embora o aplicativo DB2 possa não tirar vantagem de SQL dinâmico, o processamento de consulta federada de consultas remotas utiliza com exclusividade o SQL dinâmico.

Resolvendo o Erro `sp_helpindex`

O sistema federado depende de um dos procedimentos armazenados do catálogo do Sybase, `sp_helpindex`. Se você receber o seguinte erro de SQL, os procedimentos armazenados do catálogo do Sybase poderão não ser instalados no servidor Sybase.

```
SQL0204N "sp_helpindex" é um nome indefinido.
```

Peça ao administrador do Sybase para instalar os procedimentos armazenados do catálogo no servidor Sybase.

Related tasks:

- “Adicionando Origens de Dados Sybase a Servidores Federados” na página 79

Related reference:

- “db2set - DB2 Profile Registry Command” em *Command Reference*

Capítulo 8. Configurando o Acesso às Origens de Dados do Microsoft SQL Server

Este capítulo explica como configurar seu servidor federado para acessar dados armazenados em bancos de dados do Microsoft SQL Server.

Este capítulo lista as tarefas que precisam ser executadas, mostra exemplos das instruções SQL que precisam ser emitidas e fornece informações sobre ajuste e resolução de problemas que você pode utilizar ao fazer a configuração das origens de dados do Microsoft SQL Server.

Adicionando Origens de Dados Microsoft SQL Server a Servidores Federados

A configuração do servidor federado para acessar origens de dados Microsoft SQL Server envolve o fornecimento de informações ao servidor federado sobre as origens de dados Microsoft e objetos SQL Server que você deseja acessar.

Você pode configurar o acesso a origens de dados Microsoft SQL Server através do Centro de Controle do DB2, através do Centro de Comandos do DB2 ou através do processador da linha de comandos do DB2.

A vantagem de utilizar o Centro de Controle do DB2 é que não é necessário digitar cada instrução e comando. O Centro de Controle do DB2 fornece o método mais fácil de configurar rapidamente o acesso a origens de dados Microsoft SQL Server. No entanto, você não pode utilizar o Centro de Controle do DB2 para iniciar as seguintes tarefas de configuração:

- Testar a conexão com o servidor Microsoft SQL Server para validar a definição de servidor e mapeamentos de usuários
- Incluir ou eliminar opções de coluna

Pré-requisitos:

- Acesse o Centro de Comandos do DB2 ou o processador da linha de comandos do DB2.
- Um servidor federado e um banco de dados que são configurados para acessar origens de dados Microsoft SQL Server.
- O driver ODBC do Microsoft SQL Server que está instalado e configurado no servidor federado.

- A configuração apropriada de variáveis de ambiente do sistema, variáveis db2dj.ini e variáveis DB2 Profile Registry (db2set). As variáveis são: DJXODBCTRACE, DJX_ODBC_LIBRARY_PATH, ODBCINI, DB2LIBPATH e DB2ENVLIST.

Procedimento:

Para adicionar uma origem de dados Microsoft SQL Server a um servidor federado:

1. Prepare o servidor federado e o banco de dados federado.
 - No Windows, confirme se o DSN do Sistema ODBC está configurado corretamente e teste a conexão com o servidor remoto Microsoft SQL Server.
 - Em sistemas UNIX, atualize ou crie um arquivo odbc.ini e teste a conexão com o servidor remoto Microsoft SQL Server.
2. Registre o wrapper.
3. Registre a definição de servidor.
4. Crie os mapeamentos do usuário.
5. Teste a conexão com o servidor remoto Microsoft SQL Server.
6. Registre pseudônimos para tabelas e exibições do Microsoft SQL Server.

Related concepts:

- “Trilha Rápida para Configurar Origens de Dados” na página 1

Related tasks:

- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Preparando o Servidor Federado e o Banco de Dados para Acessarem Origens de Dados Microsoft SQL Server” na página 96

Preparando o Servidor Federado e o Banco de Dados para Acessarem Origens de Dados Microsoft SQL Server

Preparar o servidor federado e o banco de dados para acessarem origens de dados Microsoft SQL Server faz parte da maior tarefa de adicionar origens de dados Microsoft SQL Server as servidores federados.

As etapas que precisam ser seguidas para preparar o servidor federado e o banco de dados para acessarem origens de dados Microsoft SQL Server dependem do sistema operacional que está em execução no servidor federado.

Procedimento:

Para preparar o servidor e o banco de dados federados:

No Windows:

1. Verifique se o ODBC System DSN está definido para conectar-se à origem de dados Microsoft SQL Server, marcando esta definição no Painel de Controle.
 - a. No menu **Iniciar**, abra o Painel de Controle.
 - b. Dê um clique duplo em **Origens de Dados ODBC** para exibir a janela Administrador da Origem de Dados ODBC.
 - c. Clique na guia System DSN e localize a entrada para o servidor remoto Microsoft SQL Server.

A entrada é o valor que será utilizado para a opção de servidor NODE quando você registrar o servidor no banco de dados federado.
2. Na janela Administrador da Origem de Dados ODBC, selecione **Configurar** para testar a conexão a partir do ODBC Systems DSN com a origem de dados Microsoft SQL Server. Como alternativa, você pode testar a conexão utilizando a ferramenta de consulta do Microsoft SQL Server.

Em Sistemas UNIX:

1. Verifique se o arquivo `odbc.ini` está atualizado (ou criado, se necessário) no servidor federado.

Recomendação: Coloque o arquivo `odbc.ini` ou uma cópia dele no diretório pessoal do proprietário da instância do DB2.
2. Verifique se o caminho para `odbc.ini` está na variável de ambiente `ODBCINI`.
3. Verifique se o link simbólico apropriado foi criado:
 - No HP-UX, o link simbólico é de `/usr/exe/libodbcinst.sl` a `$DJX_ODBC_LIBRARY_PATH/libodbcinst.sl`.
 - No Linux, o link simbólico é de `/usr/local/locale` a `$DJX_ODBC_LIBRARY_PATH/./locale`.
 - No Solaris Operating Environment, o link simbólico é de `$HOME/sql1lib/locale` a `$DJX_ODBC_LIBRARY_PATH/./locale`. `$HOME` é o diretório pessoal do proprietário da instância do DB2.
4. Teste a conexão do servidor federado com a origem de dados Microsoft SQL Server utilizando a ferramenta **demoodbc** do DataDirect Connect ODBC.
 - a. Em um prompt de comandos do sistema operacional, emita o seguinte comando:

```
export ODBCINI=$HOME/.odbc.ini
```
 - b. Execute o script `/opt/odbc/odbc.sh`. Este script configura várias variáveis de ambiente operacionais específicas.

- c. Teste a conexão com a origem de dados Microsoft SQL Server utilizando a ferramenta **demoodbc** do DataDirect Connect ODBC. A ferramenta **demoodbc** está localizada no subdiretório /demo das bibliotecas do Connect ODBC.

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper Microsoft SQL Server.

Related tasks:

- “Registrando o Wrapper Microsoft SQL Server” na página 98

Registrando o Wrapper Microsoft SQL Server

O registro do wrapper Microsoft SQL Server faz parte da maior tarefa de adicionar origens de dados Microsoft SQL Server a servidores federados.

Para especificar o wrapper que será utilizado para acessar origens de dados Microsoft SQL Server, emita a instrução CREATE WRAPPER. O DB2 Information Integrator inclui dois wrappers para o Microsoft SQL Server. O wrapper utilizado depende do sistema operacional do servidor federado.

- Em sistemas UNIX, o nome do wrapper padrão é MSSQLODBC3 para o driver DataDirect Connect ODBC 3.7 (ou posterior).
- No Windows, o nome do wrapper padrão é DJXMSSQL3 para o driver ODBC 3.0 (ou posterior).

Procedimento:

Para especificar o wrapper que deseja utilizar para acessar origens de dados Microsoft SQL Server, emita a instrução CREATE WRAPPER.

Por exemplo, no Windows NT e Windows 2000:

```
CREATE WRAPPER DJXMSSQL3
```

Recomendação: Utilize um dos nomes de wrappers padrão (DJXMSSQL3 ou MSSQLODBC3). Ao registrar o wrapper utilizando um dos nomes padrão, o servidor federado utiliza automaticamente o nome da biblioteca padrão que está associada a esse wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, será possível substituir o nome do wrapper padrão por um nome de sua escolha. Se você utilizar um nome diferente de um dos nomes padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER. Suponha que você tenha um servidor federado em execução no AIX e decida utilizar um nome de wrapper que não seja um dos nomes padrão. A instrução CREATE WRAPPER que precisa ser emitida é:

```
CREATE WRAPPER mywrapper LIBRARY 'libdb2mssql3.a'
```

O valor *mywrapper* é o nome dado ao wrapper em vez de utilizar o nome do wrapper padrão.

Os nomes de bibliotecas de wrapper para o Microsoft SQL Server são:

Tabela 18. Nomes de bibliotecas do wrapper Microsoft SQL Server

| Sistema operacional no servidor federado | Nome da biblioteca do wrapper |
|--|-------------------------------|
| AIX | libdb2mssql3.a |
| HP-UX | libdb2mssql3.sl |
| Linux | libdb2mssql3.so |
| Ambiente Operacional Solaris | libdb2mssql3.so |
| Windows | db2mssql3.dll |

A próxima tarefa nesta seqüência de tarefas é registrar definições de servidor para uma origem de dados Microsoft SQL Server.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Microsoft SQL Server” na página 99

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando Definições de Servidor para uma Origem de Dados Microsoft SQL Server

O registro de definições de servidor para uma origem de dados Microsoft SQL Server faz parte da maior tarefa de adicionar origens de dados Microsoft SQL Server a servidores federados.

No banco de dados federado, você deve definir cada servidor remoto Microsoft SQL Server que deseja acessar. Primeiro, você deve localizar o nome do nó do servidor remoto Microsoft SQL Server e, em seguida, utilizar este nome de nó ao registrar a definição de servidor, emitindo a instrução CREATE SERVER.

Procedimento:

Para registrar uma definição de servidor para uma origem de dados Microsoft SQL Server:

1. Localize o nome do nó.

- Se seu servidor federado estiver utilizando o Windows NT ou Windows 2000, o NODE será o nome do DSN do Sistema especificado para o servidor remoto Microsoft SQL Server que está sendo acessado.
- Se seu servidor federado estiver utilizando AIX, HP-UX, Linux ou Solaris Operating Environment, o NODE será definido no arquivo .odbc.ini.

A seguir está um exemplo de um arquivo .odbc.ini no AIX.

Exemplo de arquivo .odbc.ini no AIX:

```
rawilson=MS SQL Server 7.0
medusa=MS SQL Server 7.0
[rawilson]
Driver=/opt/odbc/lib/ivmsss16.so
Description=MS SQL Server Driver for AIX
Address=9.112.30.39,1433
[medusa]
Driver=/opt/odbc/lib/ivmsss16.so
Description=MS SQL Server Driver for AIX
Address=9.112.98.123,1433
```

Na parte superior do arquivo .odbc.ini, existe uma seção rotulada [ODBC Data Sources] que lista os nós. Cada um dos nós possui uma seção [node_name] que descreve cada nó.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dado Microsoft SQL Server.

2. Emita a instrução CREATE SERVER.

Por exemplo:

```
CREATE SERVER server_name TYPE MSSQLSERVER VERSION 7.0 WRAPPER djxmssql3
OPTIONS (NODE 'sqlnode', DBNAME 'mssdb')
```

Após a definição de servidor ter sido criada, utilize a instrução ALTER SERVER para incluir ou eliminar opções do servidor.

A próxima tarefa nesta seqüência de tarefas é criar um mapeamento de usuário para uma origem de dados Microsoft SQL Server.

Related tasks:

- “Criando um Mapeamento de Usuário para uma Origem de Dados Microsoft SQL Server” na página 103

Related reference:

- “ALTER SERVER statement” no *SQL Reference, Volume 2*
- “CREATE SERVER statement” no *SQL Reference, Volume 2*

- Apêndice B, “Opções do Servidor para Sistemas Federados”, na página 391
- “Instrução CREATE SERVER - Exemplos para o Wrapper Microsoft SQL Server” na página 101

Instrução CREATE SERVER - Exemplos para o Wrapper Microsoft SQL Server

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE SERVER para registrar servidores para o wrapper Microsoft SQL Server. Este tópico inclui um exemplo completo, que mostra como registrar um servidor com parâmetros requeridos, e um exemplo com opções de servidor adicionais.

Exemplo completo:

O exemplo a seguir mostra como registrar uma definição de servidor para um wrapper Microsoft SQL Server, emitindo a instrução CREATE SERVER:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER VERSION 7.0 WRAPPER djxmssql3
    OPTIONS (NODE 'sqlnode', DBNAME 'africa')
```

sqlserver

Um nome que você atribui ao servidor remoto Microsoft SQL Server. Esse nome deve ser exclusivo. Não são permitidos nomes de servidores duplicados.

TYPE *MSSQLSERVER*

O tipo de origem de dados para o qual você está configurando o acesso. O parâmetro TYPE para os wrappers Microsoft SQL Server deve ser *MSSQLSERVER*.

VERSION *7.0*

A versão de software do servidor do banco de dados Microsoft SQL Server que você deseja acessar. As versões suportadas são 6.5, 7.0 e 2000.

WRAPPER *djxmssql3*

O nome do wrapper especificado na instrução CREATE WRAPPER.

NODE '*sqlnode*'

O nome do nó no qual o servidor remoto Microsoft SQL Server reside. Este valor faz distinção entre letras maiúsculas e minúsculas.

Embora o nome do nó seja especificado como uma opção na instrução CREATE SERVER, ele é requerido para origens de dados Microsoft SQL Server.

DBNAME '*africa*'

O nome do banco de dados que você deseja acessar. Este valor faz distinção entre letras maiúsculas e minúsculas.

Embora o nome do banco de dados esteja especificado como uma opção na instrução CREATE SERVER, ele é requerido para as origens de dados do Microsoft SQL Server.

Exemplos de opções de servidor:

Ao registrar o servidor, você pode especificar opções de servidor adicionais na instrução CREATE SERVER. Estas opções de servidor incluem opções gerais de servidor e opções de servidor específicas do Microsoft SQL Server.

O exemplo a seguir mostra como utilizar a opção de servidor COLLATING_SEQUENCE:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER VERSION 7.0 WRAPPER dxmssql3
    OPTIONS (NODE 'sqlnode', DBNAME 'africa', COLLATING_SEQUENCE 'I')
```

A opção de servidor COLLATING_SEQUENCE especifica se a origem de dados utiliza a mesma seqüência de intercalação como o servidor federado. Em um servidor do banco de dados Microsoft SQL Server que esteja executando o Windows NT ou Windows 2000, a seqüência de intercalação padrão não faz distinção entre maiúsculas e minúsculas (por exemplo, 'STEWART' e 'StewART' são consideradas iguais). Para garantir resultados corretos do servidor federado, defina a opção de servidor COLLATING_SEQUENCE como 'I'. Esta definição indica que a origem de dados Microsoft SQL Server não faz distinção entre maiúsculas e minúsculas.

Nota: O servidor federado não efetua push down de consultas se os resultados retornados das origens de dados forem diferentes dos resultados retornados durante o processamento da consulta no servidor federado. Ao definir a opção de servidor COLLATING_SEQUENCE como 'I', o servidor federado não efetua push down de consultas com dados ou expressões de cadeia e que incluem as seguintes cláusulas, predicados ou funções:

- Cláusulas GROUP BY
- Cláusulas DISTINCT
- Predicados básicos, tais como, igual a (=)
- Funções de agregação, tais como, MIN ou MAX

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Microsoft SQL Server” na página 99

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Criando um Mapeamento de Usuário para uma Origem de Dados Microsoft SQL Server

A criação de um mapeamento de usuário para uma origem de dados Microsoft SQL Server faz parte da maior tarefa de adicionar origens de dados Microsoft SQL Server a servidores federados.

Ao tentar acessar uma origem de dados Microsoft SQL Server, o servidor federado estabelece uma conexão com a origem de dados utilizando um ID do usuário e senha válidos para essa origem de dados. É necessário definir uma associação (mapeamento do usuário) entre cada ID do usuário do servidor federado e o ID do usuário e senha da origem de dados correspondente. Crie um mapeamento de usuário para cada ID do usuário que acessará o sistema federado para enviar pedidos distribuídos.

Procedimento:

Para mapear um ID do usuário local para o ID do usuário e senha do servidor remoto Microsoft SQL Server, emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR userid SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

A próxima tarefa nesta seqüência de tarefas é testar a conexão com o servidor remoto Microsoft SQL Server.

Related tasks:

- “Testando a Conexão com o Servidor Remoto Microsoft SQL Server” na página 105

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Instrução CREATE USER MAPPING - Exemplos para o Wrapper Microsoft SQL Server” na página 104

Instrução CREATE USER MAPPING - Exemplos para o Wrapper Microsoft SQL Server

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE USER MAPPING para mapear um ID do usuário do servidor federado para um ID do usuário e senha do servidor remoto Microsoft SQL Server. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o registro especial do DB2 USER com a instrução CREATE USER MAPPING.

Exemplo completo:

O exemplo a seguir mostra como mapear um ID do usuário do servidor federado para um ID do usuário e senha do servidor remoto Microsoft SQL Server:

```
CREATE USER MAPPING FOR elizabeth SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'liz', REMOTE_PASSWORD 'abc123')
```

elizabeth

Especifica o ID do usuário local que você está mapeando para um ID do usuário definido no servidor remoto Microsoft SQL Server.

SERVER *sqlserver*

Especifica o nome do servidor remoto Microsoft SQL Server definido na instrução CREATE SERVER.

REMOTE_AUTHID '*liz*'

Especifica o ID do usuário no servidor remoto Microsoft SQL Server para o qual você está mapeando *elizabeth*. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_ID como 'U' ou 'L' na instrução CREATE SERVER.

REMOTE_PASSWORD '*abc123*'

Especifica a senha associada a '*liz*'. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_PW como 'U' ou 'L' na instrução CREATE SERVER.

Exemplo especial de registro:

O exemplo a seguir mostra uma instrução CREATE USER MAPPING que inclui o registro especial USER:

```
CREATE USER MAPPING FOR USER SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'liz', REMOTE_PASSWORD 'abc123')
```

Você pode utilizar o registro especial USER 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 REMOTE_AUTHID do usuário.

Related tasks:

- “Criando um Mapeamento de Usuário para uma Origem de Dados Microsoft SQL Server” na página 103

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Testando a Conexão com o Servidor Remoto Microsoft SQL Server

O teste da conexão com o servidor remoto Microsoft SQL Server faz parte da maior tarefa de adicionar origens de dados Microsoft SQL Server a servidores federados.

Você pode testar a conexão com o servidor remoto Microsoft SQL Server utilizando a definição de servidor e mapeamentos de usuários definidos.

Procedimento:

Para testar a conexão:

1. Abra uma sessão de passagem para emitir uma instrução SQL SELECT nas tabelas do sistema Microsoft SQL Server.

Por exemplo:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM dbo.sysobjects
SET PASSTHRU RESET
```

Se a instrução SQL SELECT retornar uma contagem, a definição de servidor e o mapeamento do usuário estarão configurados adequadamente.

2. Se a instrução SQL SELECT retornar um erro, é possível que seja necessário:
 - Verificar o servidor remoto Microsoft SQL Server para certificar-se de que ele tenha sido iniciado.
 - Verificar o servidor remoto Microsoft SQL Server para certificar-se de que ele esteja configurado para conexões de entrada.
 - Verificar o mapeamento de usuário para certificar-se de que as definições para as opções REMOTE_AUTHID e REMOTE_PASSWORD

sejam válidas para as conexões com o servidor remoto Microsoft SQL Server. Alterar o mapeamento de usuário ou criar outro mapeamento, conforme necessário.

- Verificar os drivers ODBC no servidor federado DB2 para certificar-se de que eles estejam instalados e configurados corretamente para conectar-se ao servidor remoto Microsoft SQL Server.
- Verificar as definições das variáveis federadas do DB2 para verificar se estão corretas para o servidor remoto Microsoft SQL Server. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis db2dj.ini e a variável DB2 Profile Registry (db2set).
- Verifique sua definição de servidor. Se necessário, elimine-a e crie-a novamente.

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para tabelas e exibições do Microsoft SQL Server.

Related concepts:

- “Definições do Servidor e Opções do Servidor” na página 10

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Registrando Pseudônimos para Tabelas e Exibições do Microsoft SQL Server” na página 106

Related reference:

- “ALTER USER MAPPING statement” no *SQL Reference, Volume 2*

Registrando Pseudônimos para Tabelas e Exibições do Microsoft SQL Server

O registro de pseudônimos para tabelas e exibições do Microsoft SQL Server faz parte da maior tarefa de adicionar origens de dados Microsoft SQL Server a servidores federados.

Para cada servidor remoto Microsoft SQL Server definido, registre um pseudônimo para cada tabela ou exibição que deseja acessar. Utilize estes pseudônimos, em vez dos nomes de objetos de origem de dados, quando consultar os servidores remotos Microsoft SQL Server.

O banco de dados federado depende das estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento da consulta. Essas estatísticas

são coletadas ao registrar um pseudônimo para um objeto da origem de dados utilizando a instrução `CREATE NICKNAME`. O banco de dados federado verifica a presença do objeto na origem de dados e tenta coletar os dados estatísticos existentes da origem de dados. As informações úteis ao otimizador são lidas a partir dos catálogos de origem de dados e colocadas no catálogo global no servidor federado. Como algumas informações do catálogo de origem de dados, ou todas elas, podem ser utilizadas pelo otimizador, atualize as estatísticas (utilizando o comando de origem de dados equivalente ao comando `RUNSTATS` do DB2) na origem de dados antes de registrar um pseudônimo.

Procedimento:

Para registrar um pseudônimo, emita a instrução `CREATE NICKNAME`.

Por exemplo:

```
CREATE NICKNAME mss_name FOR sqlserver."remote_schema"."remote.table"
```

Os pseudônimos podem ter até 128 caracteres.

Repita esta etapa para cada tabela ou exibição do Microsoft SQL Server para a qual você deseja criar um pseudônimo.

Quando você criar o pseudônimo, o DB2 utiliza a conexão para consultar as tabelas do catálogo da origem de dados (o Microsoft SQL Server refere-se a estas tabelas como tabelas do sistema). Esta consulta testa a conexão com a origem de dados utilizando o pseudônimo. Se a conexão não funcionar, você receberá uma mensagem de erro.

Related reference:

- “`RUNSTATS` Command” na publicação *Command Reference*
- “`CREATE NICKNAME` statement” no *SQL Reference, Volume 2*
- “Instrução `CREATE NICKNAME` - Exemplos para o Wrapper Microsoft SQL Server” na página 107

Instrução `CREATE NICKNAME` - Exemplos para o Wrapper Microsoft SQL Server

Este tópico fornece um exemplo que mostra como utilizar a instrução `CREATE NICKNAME` para registrar um pseudônimo para uma tabela ou exibição do Microsoft SQL Server que você deseja acessar.

Este exemplo mostra como especificar um objeto remoto para o servidor Microsoft SQL Server com o qual o pseudônimo é atribuído:

```
CREATE NICKNAME cust_africa FOR sqlserver.customers.egypt
```

cust_africa

Um pseudônimo exclusivo para a tabela ou exibição do Microsoft SQL Server.

Nota: O pseudônimo é um nome com duas partes que inclui o esquema e o pseudônimo. Se omitir o esquema ao registrar o pseudônimo, o esquema do pseudônimo será o ID de autenticação do usuário que está criando o pseudônimo.

sqlserver.customers.egypt

Um identificador de três partes para o objeto remoto.

- *sqlserver* é o nome atribuído ao servidor do banco de dados Microsoft SQL Server na instrução CREATE SERVER.
- *customers* é o nome do esquema remoto ao qual pertence a tabela ou exibição.
- *egypt* é o nome da tabela ou exibição remota que você deseja acessar.

O servidor federado dobra nomes de esquemas e tabelas do Microsoft SQL Server para maiúsculas, a menos que você coloque os nomes entre aspas.

Related tasks:

- “Registrando Pseudônimos para Tabelas e Exibições do Microsoft SQL Server” na página 106

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*

Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Microsoft SQL Server

Depois de fazer a configuração para origens de dados Microsoft SQL Server, talvez você queira modificar a configuração para aprimorar o desempenho. Por exemplo, talvez você queira definir a variável do registro de perfil DB2_DJ_COMM para aprimorar o desempenho quando o servidor federado acessa a origem de dados Microsoft SQL Server.

Aprimorando o Desempenho Definindo a Variável DB2_DJ_COMM (UNIX)

Se perceber que demora muito tempo para acessar o servidor remoto Microsoft SQL Server, você poderá aprimorar o desempenho, definindo a variável de registro do perfil do DB2 DB2_DJ_COMM. Ao definir a variável DB2_DJ_COMM, o servidor federado carregará o wrapper durante a inicialização em vez de quando você tenta acessar a origem de dados.

Procedimento:

Para definir a variável DB2_DJ_COMM:

1. Defina a variável DB2_DJ_COMM para o wrapper da biblioteca correspondente ao wrapper especificado. Suponha que o servidor federado esteja executando o AIX e o wrapper que você está utilizando seja o MSSQLODBC3. O comando para definir a variável DB2_DJ_COMM é:

```
db2set DB2_DJ_COMM='libdb2mssql3.a'
```

A tabela a seguir lista os nomes de bibliotecas válidos por sistema operacional suportado.

Tabela 19. Nomes de bibliotecas do wrapper Microsoft SQL Server

| Sistema operacional no servidor federado | Nomes de bibliotecas do wrapper MSSQLODBC3 | Nomes de bibliotecas do wrapper DJXMSSQL3 |
|--|--|---|
| AIX | libdb2mssql3.a | nenhuma |
| HP-UX | libdb2mssql3.sl | nenhuma |
| Linux | libdb2mssql3.so | nenhuma |
| Ambiente Operacional Solaris | libdb2mssql3.so | nenhuma |
| Windows NT e Windows 2000 | nenhuma | db2mssql3.dll |

2. Emita os comandos a seguir, para reciclar a instância do DB2:

```
db2stop  
db2start
```

Ao reciclar a instância do DB2, assegure que ela aceite alterações de variáveis feitas.

Obtendo Rastreios ODBC

Se você estiver tendo problemas quando acessar a origem de dados, poderá obter as informações de rastreamento do ODBC para analisar e resolver esses problemas. A ativação de um rastreamento prejudica o desempenho do sistema. Portanto, você deve desativar o rastreamento depois de resolver os problemas.

Em servidores federados Windows, utilize a ferramenta de rastreamento que é fornecida pelo Administrador da Origem de Dados ODBC para assegurar que o rastreamento do ODBC funcione corretamente.

Em servidores federados UNIX, defina a variável DJXODBCTRACE no arquivo db2dj.ini. Por exemplo:

```
DJXODBCTRACE=/home/user1/trace_dir/filename.xxx
```

Também é necessário ativar o rastreamento para o arquivo .odbc.ini. Por exemplo, suponha que você esteja utilizando o driver DataDirect ODBC 3.x. Localize o

exemplo do arquivo .odbc.ini no diretório do cliente. Este arquivo contém uma amostra do que é necessário para arquivos de rastreo:

```
[ODBC]
Trace=0
TraceFile=/home/user1/trace_dir/filename.xxx
TraceDll==/opt/odbc/lib/odbcdrac.so
InstallDir=/opt/odbc
```

A primeira linha é definida como Trace=0 quando o rastreo está OFF e esta primeira linha é definida como Trace=1 quando o rastreo está ON. O TraceFile deve apontar para um caminho e nome do arquivo aos quais a instância tenha acesso de gravação. Este caminho e nome de arquivo também devem corresponder à linha que é colocada no arquivo db2dj.ini, DJXODBCTRACE=/home/user1/trace_dir/filename.xxx

Related tasks:

- “Adicionando Origens de Dados Microsoft SQL Server a Servidores Federados” na página 95

Related reference:

- “db2set - DB2 Profile Registry Command” em *Command Reference*

Capítulo 9. Configurando o Acesso às Origens de Dados do ODBC

Este capítulo explica como configurar seu servidor federado para acessar dados armazenados em origens de dados do ODBC.

Este capítulo lista as tarefas que precisam ser executadas, mostra exemplos das instruções SQL que precisam ser emitidas e fornece informações sobre ajuste e resolução de problemas que você pode utilizar ao fazer a configuração das origens de dados do ODBC.

Adicionando Origens de Dados ODBC a Servidores Federados

O DB2 Information Integrator fornece wrappers que suportam origens de dados específicas que são acessadas através da API ODBC. Exemplos destas origens de dados incluem Oracle, Microsoft SQL Server e Microsoft Excel. Você terá melhor desempenho se utilizar os wrappers projetados especificamente para essas origens de dados. As origens de dados que são acessadas através da API ODBC são referidas neste texto como origens de dados ODBC.

Utilize o wrapper ODBC para acessar qualquer origem de dados que tenha um driver ODBC mas não seja suportada por wrappers de origens de dados específicas que estão incluídas no DB2 Information Integrator.

O wrapper ODBC suporta o ODBC Versão 3.x.

A configuração do servidor federado para acessar origens de dados ODBC envolve o fornecimento de informações ao servidor federado sobre as origens de dados ODBC e objetos que você deseja acessar.

Você pode configurar o acesso a origens de dados ODBC através do Centro de Controle do DB2, através do Centro de Comandos do DB2 ou através do processador da linha de comandos do DB2.

A vantagem de utilizar o Centro de Controle do DB2 é que não é necessário digitar cada instrução e comando. O Centro de Controle do DB2 fornece o método mais fácil de configurar rapidamente o acesso a origens de dados ODBC.

Você pode utilizar o wrapper ODBC em servidores federados que utilizam os seguintes sistemas operacionais:

- AIX
- HP-UX
- Sistemas operacionais Linux no Intel
- Ambiente Operacional Solaris
- Windows NT, Windows 2000, Windows .NET

Pré-requisitos:

- Acesse o Centro de Comandos do DB2 ou o processador da linha de comandos do DB2.
- Um servidor federado e um banco de dados que são configurados para acessar origens de dados ODBC.
- O driver ODBC instalado e configurado no servidor federado.
- A configuração apropriada de variáveis de ambiente do sistema, variáveis db2dj.ini e variáveis DB2 Profile Registry (db2set). Verifique a documentação do fornecedor para as variáveis requeridas para o cliente ODBC. A variável LIBPATH pode ser requerida.

Restrições:

- O wrapper ODBC não suporta as seguintes funções e instruções:
 - Instruções LOCK TABLE em pseudônimos
 - Recursos reprovados no ODBC Versão 3.x
 - Drivers X/Open ou SQL/CLI
 - Pseudônimos de procedimentos armazenados
 - Execução de atomicidade em nível de instrução utilizando instruções de pontos de salvamento remotas
 - Clientes de 64 bits
- As instruções posicionadas UPDATE e DELETE e algumas complexas, as instruções pesquisadas UPDATE e DELETE em um pseudônimo falharão se um índice exclusivo de colunas não anuláveis não existir no pseudônimo ou em sua tabela remota correspondente.
- O wrapper ODBC suporta operações de leitura e gravação com a maioria das origens de dados.

Procedimento:

Para incluir uma origem de dados ODBC em um servidor federado:

1. Prepare o servidor federado e o banco de dados federado.
2. Registre o wrapper.
3. Registre a definição de servidor.
4. Crie os mapeamentos do usuário.

5. Teste a conexão com a origem de dados ODBC.
6. Registre pseudônimos para tabelas e exibições de origens de dados ODBC.

Related concepts:

- “Trilha Rápida para Configurar Origens de Dados” na página 1

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Preparando o Servidor Federado e o Banco de Dados para Acessarem Origens de Dados através do ODBC” na página 113

Preparando o Servidor Federado e o Banco de Dados para Acessarem Origens de Dados através do ODBC

A preparação do servidor federado e do banco de dados para acessarem origens de dados através do ODBC faz parte da maior tarefa de adicionar origens de dados ODBC a servidores federados.

As etapas que precisam ser seguidas para preparar o servidor federado e o banco de dados para acessarem origens de dados através do ODBC dependem do sistema operacional que está em execução no servidor federado.

Nota: O driver ODBC e o sistema operacional que estão sendo utilizados possuem localizações de caminhos de bibliotecas exclusivas.

Procedimento:

Para preparar o servidor e o banco de dados federados:

No Windows:

1. Verifique se o DSN do Sistema ODBC está definido para conectar-se à origem de dados ODBC. Você pode utilizar o Administrador da Origem de Dados ODBC para configurar o DSN. Verifique esta definição no Painel de Controle.
 - a. No menu **Iniciar**, abra o Painel de Controle.
 - b. Dê um clique duplo em **Origens de Dados ODBC** para acessar o gerenciador de dispositivos ODBC.
 - c. Clique na guia DSN do Sistema para confirmar se o DSN do Sistema definido para o driver ODBC aparece na lista.

O nome do nó para a origem de dados ODBC deve ser definido no DSN do Sistema.

2. Na janela Administrador da Origem de Dados ODBC, selecione **Configurar** para testar a conexão do DSN de Sistemas ODBC com a origem de dados ODBC.

Em Sistemas UNIX:

Consulte a documentação do fornecedor do cliente ODBC para obter instruções sobre como configurar o cliente ODBC.

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper ODBC.

Related tasks:

- “Registrando o Wrapper ODBC” na página 114

Registrando o Wrapper ODBC

O registro do wrapper ODBC faz parte da maior tarefa de adicionar origens de dados ODBC a servidores federados.

Você deve emitir a instrução CREATE WRAPPER para registrar um wrapper ODBC.

Procedimento:

Para especificar o wrapper que deseja utilizar para acessar origens de dados ODBC, emita a instrução CREATE WRAPPER.

Por exemplo:

```
CREATE WRAPPER odbc
```

Recomendação: Utilize o nome do wrapper padrão chamado ODBC ao emitir a instrução CREATE WRAPPER. Ao registrar o wrapper que utiliza o nome padrão, o servidor federado utiliza automaticamente o nome da biblioteca padrão associado a esse wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, será possível substituir o nome do wrapper padrão por um nome de sua escolha. Se você utilizar um nome que seja diferente do padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER. Suponha que você tenha um servidor federado em execução no AIX e decida utilizar um nome de wrapper que não seja o padrão. Um exemplo de instrução CREATE WRAPPER que precisa ser emitida é:

```
CREATE WRAPPER mywrapper  
LIBRARY 'libdb2rcodbc.a' OPTIONS (MODULE '/usr/lib/odbc.a')
```

MODULE *'/usr/lib/odbc.a'* é o caminho completo da biblioteca que contém o Gerenciador de Dispositivos ODBC.

É necessário registrar o wrapper ODBC apenas uma vez, independentemente do número de origens de dados ODBC que você planeja acessar. Especifique a localização da origem de dados quando registrar a definição de servidor. Especifique o objeto de origem de dados exato quando registrar o pseudônimo.

A próxima tarefa nesta seqüência de tarefas é registrar definições de servidor para uma origem de dados ODBC.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados ODBC” na página 116

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*
- “Instrução CREATE WRAPPER - Exemplos para o Wrapper ODBC” na página 115

Instrução CREATE WRAPPER - Exemplos para o Wrapper ODBC

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE WRAPPER para registrar wrappers para origens de dados ODBC.

Exemplo para sistemas UNIX:

O exemplo a seguir mostra como registrar wrapper, emitindo a instrução CREATE WRAPPER em um sistema operacional UNIX:

```
CREATE WRAPPER odbc OPTIONS (MODULE '/usr/lib/odbc.so')
```

Neste exemplo, *odbc* é o nome atribuído ao wrapper que está sendo registrado no banco de dados federado. MODULE *'/usr/lib/odbc.so'* é o caminho completo da biblioteca que contém o Gerenciador de Dispositivos ODBC.

Você deve especificar a opção MODULE em sistemas operacionais UNIX. No Windows, a opção MODULE assume como padrão *'odbc32.dll'*.

Exemplo para Windows:

O exemplo a seguir mostra como registrar um wrapper, emitindo a instrução CREATE WRAPPER em um sistema operacional Windows:

```
CREATE WRAPPER odbc LIBRARY 'db2rcodbc.dll'
```

Neste exemplo, *odbc* é o nome atribuído ao wrapper que está sendo registrado no banco de dados federado. `LIBRARY 'db2rcodbc.dll'` é o nome da biblioteca para o wrapper ODBC.

A tabela a seguir lista os nomes de bibliotecas de wrapper para ODBC por sistema operacional:

Tabela 20. Nomes de bibliotecas do wrapper ODBC

| Sistema operacional em seu servidor federado | Nomes de bibliotecas do wrapper ODBC |
|--|--------------------------------------|
| AIX | libdb2rcodbc.a |
| HP-UX | libdb2rcodbc.sl |
| Linux | libdb2rcodbc.so |
| Ambiente Operacional Solaris | libdb2rcodbc.so |
| Windows | db2rcodbc.dll |

Related tasks:

- “Registrando o Wrapper ODBC” na página 114

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando Definições de Servidor para uma Origem de Dados ODBC

O registro de definições de servidor para uma origem de dados ODBC faz parte da maior tarefa de adicionar origens de dados ODBC a servidores federados.

No banco de dados federado você deve definir cada servidor da origem de dados ODBC que deseja acessar.

Procedimento:

Para registrar uma definição de servidor para uma origem de dados ODBC:

Emita a instrução `CREATE SERVER`.

Por exemplo:

```
CREATE SERVER server_name TYPE odbc
VERSION 3.0 WRAPPER odbc_wrapper
      OPTIONS (NODE 'node_name')
```

Embora NODE esteja especificado como um opcional na instrução CREATE SERVER, ele é obrigatório para origens de dados ODBC.

Após a definição de servidor ter sido criada, utilize a instrução ALTER SERVER para incluir ou eliminar opções do servidor.

A próxima tarefa nesta seqüência de tarefas é criar um mapeamento de usuário para uma origem de dados ODBC.

Related tasks:

- “Criando um Mapeamento de Usuário para uma Origem de Dados ODBC” na página 119

Related reference:

- “ALTER SERVER statement” no *SQL Reference, Volume 2*
- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- Apêndice B, “Opções do Servidor para Sistemas Federados”, na página 391
- “Instrução CREATE SERVER - Exemplos de Wrapper ODBC” na página 117

Instrução CREATE SERVER - Exemplos de Wrapper ODBC

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE SERVER para registrar servidores para o wrapper ODBC. Este tópico inclui um exemplo completo, que mostra como registrar um servidor com parâmetros requeridos e um exemplo com opções de servidor adicionais.

Exemplo completo:

O exemplo a seguir mostra como registrar uma definição de servidor para um wrapper ODBC, emitindo a instrução CREATE SERVER:

```
CREATE SERVER server_name TYPE odbc
VERSION 3.0 WRAPPER odbc_wrapper
      OPTIONS (NODE 'node_name', DBNAME 'venice')
```

server_name

Um nome atribuído ao servidor da origem de dados ODBC. Esse nome deve ser exclusivo. Não são permitidos nomes de servidores duplicados.

TYPE *odbc*

Especifica o tipo de origem de dados para o qual você está configurando o acesso. Para o wrapper ODBC, o tipo de servidor deve ser *odbc*.

VERSION 3.0

A versão do cliente ODBC que você deseja acessar. Todos os releases do ODBC padrão Versão 3 são suportados.

WRAPPER *odbc_wrapper*

O nome do wrapper especificado na instrução CREATE WRAPPER.

NODE '*node_name*'

O nome do nó (o nome DSN do sistema) que foi atribuído à origem de dados ODBC quando o DSN foi definido. Este valor faz distinção entre letras maiúsculas e minúsculas. No Windows, este valor deve ser o nome de um DSN do sistema na janela Administração de Dados ODBC. No UNIX, consulte a documentação do fornecedor do cliente ODBC para obter informações sobre o valor a ser utilizado.

Embora o NODE esteja especificado como uma opção na instrução CREATE SERVER, ele é obrigatório para origens de dados ODBC.

DBNAME '*venice*'

O nome do banco de dados que você deseja acessar. Este valor faz distinção entre letras maiúsculas e minúsculas.

Exemplo de opções de servidor:

O exemplo a seguir mostra como utilizar as opções de servidor DB2_TABLE_QUOTE_CHAR e DB2_ID_QUOTE_CHAR.

Algumas origens de dados ODBC (por exemplo, MySQL) não podem processar aspas nos nomes de tabelas e nos nomes de colunas em instruções SQL. Para acessar essas origens de dados, é necessário fornecer as seguintes opções de servidor na instrução CREATE SERVER:

- DB2_TABLE_QUOTE_CHAR ' '
- DB2_ID_QUOTE_CHAR ' '

Por exemplo:

```
CREATE SERVER mysql_server TYPE odbc  
VERSION 3.0 WRAPPER odbc_wrapper  
OPTIONS (NODE 'mysql', DB2_TABLE_QUOTE_CHAR ' ',  
DB2_ID_QUOTE_CHAR ' ')
```

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados ODBC” na página 116

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Criando um Mapeamento de Usuário para uma Origem de Dados ODBC

A criação de um mapeamento de usuário para uma origem de dados ODBC faz parte da maior tarefa de adicionar origens de dados ODBC a servidores federados.

Ao tentar acessar uma origem de dados ODBC, o servidor federado estabelece uma conexão com a origem de dados utilizando um ID do usuário e senha válidos para essa origem de dados. É necessário definir uma associação (mapeamento do usuário) entre cada ID do usuário do servidor federado e o ID do usuário e senha da origem de dados correspondente. Crie um mapeamento de usuário para cada ID do usuário que acessará o sistema federado para enviar pedidos distribuídos.

Procedimento:

Para mapear um ID do usuário local para o ID do usuário e senha da origem de dados ODBC, emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR userid SERVER server_name
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

A próxima tarefa nesta seqüência de tarefas é testar a conexão com o servidor da origem de dados ODBC.

Related tasks:

- “Testando a Conexão com o Servidor da Origem de Dados ODBC” na página 120

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Instrução CREATE USER MAPPING - Exemplos para o Wrapper ODBC” na página 119

Instrução CREATE USER MAPPING - Exemplos para o Wrapper ODBC

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE USER MAPPING para mapear um ID do usuário local para um ID do usuário e senha da origem de dados ODBC. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o registro especial do DB2 USER com a instrução CREATE USER MAPPING.

Exemplo completo:

O exemplo a seguir mostra como mapear um ID do usuário local para um ID do usuário e senha da origem de dados ODBC:

```
CREATE USER MAPPING FOR arturo SERVER server_name
    OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue')
```

arturo Especifica o ID do usuário local que você está mapeando para um ID do usuário definido na origem de dados ODBC.

server_name

Especifica o nome da origem de dados ODBC definida na instrução CREATE SERVER.

'art' Especifica o ID do usuário na origem de dados ODBC para o qual você está mapeando *arturo*. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_ID como 'U' ou 'L' na instrução CREATE SERVER.

'red4blue'

Especifica a senha associada a *'art'*. Utilize aspas simples para preservar o tipo de letra deste valor, a menos que defina a opção de servidor FOLD_PW como 'U' ou 'L' na instrução CREATE SERVER.

Exemplo especial de registro:

O exemplo a seguir mostra uma instrução CREATE USER MAPPING que inclui o registro especial USER:

```
CREATE USER MAPPING FOR USER SERVER server_name
    OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue')
```

Você pode utilizar o registro especial USER 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 REMOTE_AUTHID do usuário.

Related tasks:

- “Criando um Mapeamento de Usuário para uma Origem de Dados ODBC” na página 119

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Testando a Conexão com o Servidor da Origem de Dados ODBC

O teste da conexão com o servidor ODBC faz parte da maior tarefa de adicionar origens de dados ODBC a servidores federados.

Você pode testar a conexão com o servidor ODBC utilizando a definição de servidor e mapeamentos de usuários definidos.

Pré-requisitos:

A origem de dados que está sendo utilizada deve suportar sessões de passagem.

Procedimento:

Para testar a conexão:

1. Abra uma sessão de passagem para emitir uma instrução SQL SELECT nas tabelas do sistema da origem de dados ODBC.

Por exemplo:

```
SET PASSTHRU server_name
SELECT COUNT(*) FROM schema_name.table_name
SET PASSTHRU RESET
```

server_name é o nome da origem de dados ODBC definida na instrução CREATE SERVER.

schema_name é o nome do esquema na origem de dados ODBC remota. Se a origem de dados ODBC não suportar esquemas, omite o esquema da instrução.

table_name é o nome da tabela na origem de dados ODBC remota.

Se a instrução SQL SELECT retornar uma contagem, isto indica que sua definição de servidor e seus mapeamentos de usuários estão configurados corretamente.

2. Se a instrução SQL SELECT retornar um erro, é possível que seja necessário:
 - Verificar se a origem de dados está disponível.
 - Se aplicável, verificar o servidor da origem de dados para certificar-se de que ele esteja configurado para conexões de entrada.
 - Verificar o mapeamento de usuário para certificar-se de que as definições para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com a origem de dados ODBC. Alterar o mapeamento de usuário ou criar outro mapeamento, conforme necessário.
 - Verificar o driver ODBC no servidor federado DB2 para assegurar que foi instalado e configurado corretamente para conectar-se ao servidor de origem de dados ODBC. Em sistemas operacionais Windows, utilize a

ferramenta do Administrador da Origem de Dados ODBC para verificar o driver. Em sistemas operacionais UNIX, consulte a documentação do fornecedor do cliente ODBC.

- Verifique sua definição de servidor. Se necessário, elimine-a e crie-a novamente.

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para tabelas e exibições da origem de dados ODBC.

Related concepts:

- “Definições do Servidor e Opções do Servidor” na página 10

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*
- “Registrando Pseudônimos para Tabelas e Exibições de Origens de Dados ODBC” na página 122

Related reference:

- “ALTER USER MAPPING statement” no *SQL Reference, Volume 2*

Registrando Pseudônimos para Tabelas e Exibições de Origens de Dados ODBC

O registro de pseudônimos para tabelas de exibições do ODBC faz parte da maior tarefa de adicionar origens de dados ODBC a servidores federados.

Para cada servidor ODBC definido, registre um pseudônimo para cada tabela ou exibição que deseja acessar. Utilize estes pseudônimos, em vez dos nomes de objetos de origem de dados, quando consultar as origens de dados ODBC.

Além de registrar pseudônimos para tabelas e exibições da origem de dados ODBC, você também pode registrar pseudônimos para tabelas do sistema remotas.

Por exemplo, suponha que tenha definido o pseudônimo *cust_europe* para representar uma tabela do Microsoft SQL Server chamada *italy* com um nome de esquema *customers*. A instrução SQL `SELECT * FROM cust_europe` é permitida a partir do servidor federado. No entanto, a instrução `SELECT * FROM server_name."customers"."italy"` não é permitida.

Se a origem de dados ODBC não suportar esquemas, omita o esquema da instrução.

O banco de dados federado depende das estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento da consulta. Essas estatísticas são coletadas ao registrar um pseudônimo para um objeto da origem de dados utilizando a instrução CREATE NICKNAME. O banco de dados federado verifica a presença do objeto na origem de dados e tenta coletar os dados estatísticos existentes da origem de dados. As informações úteis ao otimizador são lidas a partir dos catálogos de origem de dados e colocadas no catálogo global no servidor federado. Como algumas informações do catálogo de origem de dados, ou todas elas, podem ser utilizadas pelo otimizador, atualize as estatísticas (utilizando o comando de origem de dados equivalente ao comando RUNSTATS do DB2) na origem de dados antes de registrar um pseudônimo.

Procedimento:

Para registrar um pseudônimo, emita a instrução CREATE NICKNAME.

Por exemplo:

```
CREATE NICKNAME odbc_name FOR server_name."remote_schema"."remote.table"
```

Os pseudônimos podem ter até 128 caracteres.

Repita esta etapa para cada tabela ou exibição do ODBC para a qual você deseja criar um pseudônimo.

Quando cria o pseudônimo, o DB2 utiliza a conexão para consultar as tabelas do catálogo da origem de dados. Essa consulta testa a conexão com a origem de dados ODBC utilizando o pseudônimo. Se a conexão não funcionar, você receberá uma mensagem de erro.

Related reference:

- “RUNSTATS Command” na publicação *Command Reference*
- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper ODBC” na página 123

Instrução CREATE NICKNAME - Exemplos para o Wrapper ODBC

Este tópico fornece um exemplo que mostra como utilizar a instrução CREATE NICKNAME para registrar um pseudônimo para uma tabela ou exibição do ODBC que você deseja acessar.

Este exemplo mostra como especificar um objeto remoto para a origem de dados ODBC com o qual o pseudônimo é atribuído:

```
CREATE NICKNAME cust_europe FOR server_name."customers"."italy"
```

cust_europe

Um pseudônimo exclusivo para a tabela ou exibição. O pseudônimo deve ser exclusivo dentro do esquema.

Nota: O pseudônimo é um nome com duas partes que inclui o esquema e o pseudônimo. Se omitir o esquema ao registrar o pseudônimo, o esquema do pseudônimo será o ID de autenticação do usuário que registra o pseudônimo.

server_name."customers"."italy"

Um identificador de três partes para o objeto remoto.

- *server_name* é o nome atribuído ao servidor do banco de dados ODBC na instrução CREATE SERVER.
- *customers* é o nome do esquema remoto ao qual pertence a tabela ou exibição. Se a origem de dados ODBC não suportar esquemas, omita o esquema da instrução CREATE NICKNAME.
- *italy* é o nome da tabela remota ou exibição que você deseja acessar.

Os objetos da origem de dados ODBC podem fazer distinção entre maiúsculas e minúsculas. Coloque o nome do esquema remoto e o nome da tabela remota entre aspas. De outra maneira, o DB2 dobrará estes nomes para maiúsculas.

Related tasks:

- “Registrando Pseudônimos para Tabelas e Exibições de Origens de Dados ODBC” na página 122

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*

Ajustando e Resolvendo Problemas de Configuração para Origens de Dados ODBC

Depois de fazer a configuração para origens de dados ODBC, talvez você queira modificar a configuração para aprimorar o desempenho. Por exemplo, talvez você queira definir a variável do registro de perfil DB2_DJ_COMM para aprimorar o desempenho quando o servidor federado acessar a origem de dados ODBC.

Aprimorando o Desempenho Definindo a Variável DB2_DJ_COMM

Se perceber que demora muito tempo para acessar o servidor remoto ODBC, você poderá aprimorar o desempenho, definindo a variável de registro do perfil do DB2 DB2_DJ_COMM. Quando definir a variável DB2_DJ_COMM, o servidor federado carregará o wrapper durante a inicialização em vez de quando você tenta acessar a origem de dados.

Procedimento:

Para definir a variável DB2_DJ_COMM:

1. Defina a variável DB2_DJ_COMM para o wrapper da biblioteca correspondente ao wrapper especificado.

Por exemplo:

Suponha que seu servidor federado utilize o Windows NT e o wrapper que está sendo utilizado seja ODBC_WRAPPER. O comando para definir a variável DB2_DJ_COMM é:

```
db2set DB2_DJ_COMM='db2rcodbc.dll'
```

A variável DB2_DJ_COMM é adicionada ao Registro do Windows.

A tabela a seguir lista os nomes de bibliotecas ODBC corretos por sistemas operacionais suportados.

Tabela 21. Nomes de bibliotecas do wrapper ODBC

| Sistema operacional em seu servidor federado | Nomes de bibliotecas do wrapper ODBC |
|--|---|
| AIX | libdb2rcodbc.a |
| HP-UX | libdb2rcodbc.sl |
| Linux | libdb2rcodbc.so |
| Ambiente Operacional Solaris | libdb2rcodbc.so |
| Windows NT, Windows 2000 e Windows .NET | db2rcodbc.dll, db2rcodbcF.dll, db2rcodbcU.dll |

2. Emita os comandos a seguir, para reciclar a instância do DB2:

```
db2stop  
db2start
```

Ao reciclar a instância do DB2, assegure que ela aceite alterações feitas.

Obtendo Rastreios ODBC

Se você estiver tendo problemas quando acessar a origem de dados, poderá obter as informações de rastreamento do ODBC para analisar e resolver esses problemas. A ativação de um rastreamento prejudica o desempenho do sistema. Portanto, você deve desativar o rastreamento depois de resolver os problemas.

Em servidores federados Windows, utilize a ferramenta de rastreamento fornecida pelo Administrador da Origem de Dados ODBC para assegurar que o rastreamento do ODBC funcione corretamente. Em sistemas operacionais UNIX, consulte a documentação do fornecedor do cliente ODBC.

Related tasks:

- “Adicionando Origens de Dados ODBC a Servidores Federados” na página 111

Related reference:

- “db2set - DB2 Profile Registry Command” em *Command Reference*

Capítulo 10. Configurando o Acesso às Origens de Dados do Teradata

Este capítulo explica como configurar seu servidor federado para acessar dados armazenados em bancos de dados do Teradata.

Este capítulo lista as tarefas que precisam ser executadas, mostra exemplos das instruções SQL que precisam ser emitidas e fornece informações sobre ajuste e resolução de problemas que você pode utilizar ao fazer a configuração de uma origem de dados do Teradata.

Adicionando Origens de Dados Teradata a Servidores Federados

Para acessar origens de dados Teradata a partir de um servidor federado, é necessário fornecer ao servidor federado informações sobre as origens de dados Teradata e os objetos que você deseja acessar.

Você pode configurar o acesso a origens de dados Teradata através do Centro de Controle do DB2, através do Centro de Comandos do DB2 ou através do processador da linha de comandos do DB2.

A vantagem de utilizar o Centro de Controle do DB2 é que não é necessário digitar cada instrução e comando. O Centro de Controle do DB2 fornece o método mais fácil de configurar rapidamente o acesso a origens de dados Teradata. No entanto, você não pode utilizar o Centro de Controle do DB2 para emitir instruções SQL.

Você pode adicionar um wrapper Teradata a seu servidor federado em qualquer um dos sistemas operacionais suportados:

- AIX Versão 4.3 ou posterior
- Windows NT e Windows 2000

Pré-requisitos:

- Acesse o Centro de Comandos do DB2 ou o processador da linha de comandos do DB2.
- Um servidor federado e um banco de dados que são configurados para acessar origens de dados Teradata.
- O software do cliente Teradata que suporta o Teradata Call-Level Interface, Versão 2 (CLIV2) Release 04.06 ou posterior e está instalado e configurado no servidor federado.

- Acesso a um ou mais servidores Teradata que estão executando o Teradata release V2R3 ou V2R4.
- Variáveis de ambiente do sistema, variáveis db2dj.ini e variáveis DB2 Profile Registry (db2set) que estão corretamente definidas para acessar origens de dados Teradata.

As variáveis db2dj.ini que precisam ser configuradas corretamente incluem:

- As variáveis COPLIB e COPERR (apenas AIX).

Por exemplo:

```
COPLIB=coplib_directory
COPERR=coperr_directory
```

O *coplib_directory* é o caminho completo do diretório no qual o arquivo libcliv2.so reside. O *coperr_directory* é o caminho completo do diretório no qual o arquivo errmsg.txt reside.

Por padrão, o processo de instalação coloca o arquivo libcliv2.so e o arquivo errmsg.txt no mesmo diretório. No entanto, você pode especificar durante o processo de instalação que o arquivo libcliv2.so e o arquivo errmsg.txt residem em diretórios diferentes.

- As variáveis do Teradata NETRACE e COPANOMLOG (opcionais).

Você pode ativar o rastreamento do Teradata se precisar preservar uma listagem de instruções SQL que são enviadas para o servidor Teradata.

Por exemplo:

```
NETRACE=1
COPANOMLOG=trace_file
```

O *trace_file* é o nome completo do arquivo que conterá os dados de rastreamento.

Estas variáveis ativam somente o recurso de rastreamento do Teradata e não afetam o rastreamento do DB2.

Procedimento:

Para adicionar uma origem de dados Teradata a um servidor federado:

1. Opcional: Teste a conexão com o servidor Teradata.
2. Verifique se a biblioteca Teradata está ativada para o link de tempo de execução (AIX).
3. Registre o wrapper.
4. Registre a definição de servidor.
5. Crie os mapeamentos do usuário.
6. Teste a conexão do servidor federado com o servidor Teradata.

7. Registre pseudônimos para tabelas e exibições do Teradata.

Related concepts:

- “Trilha Rápida para Configurar Origens de Dados” na página 1

Related tasks:

- “Testando a Conexão com o Servidor Teradata” na página 129
- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*

Testando a Conexão com o Servidor Teradata

O teste da conexão com o servidor Teradata faz parte da maior tarefa de adicionar origens de dados Teradata a servidores federados.

Antes de criar um wrapper, uma definição de servidor ou mapeamento de usuário, você pode testar a conexão com o servidor Teradata. Teste a conexão primeiro para verificar se o software do cliente está corretamente definido e para evitar erros ao emitir as instruções CREATE WRAPPER, CREATE SERVER e CREATE USER MAPPING.

Você pode utilizar o utilitário BTEQ (Basic Teradata Query) para enviar uma consulta SQL para verificar se pode conectar-se ao servidor Teradata. Consulte a documentação do Teradata para obter mais informações sobre o utilitário BTEQ.

Pré-requisito:

Assegure que o utilitário BTEQ e o Teradata Data Connector Application Program Interface (PIOM) foram instalados durante o processo de instalação do software de cliente Teradata.

Procedimento:

Para testar a conexão com o servidor Teradata:

1. Inicie uma sessão do utilitário BTEQ e efetue logon no servidor Teradata.
2. Emita um comando SQL para verificar se pode conectar-se com êxito ao servidor Teradata. Por exemplo:

```
select count(*) from dbc.tables;
```

Se a conexão for bem-sucedida, deverá aparecer a saída da consulta na tela. Por exemplo:

```
*** Query completed. One row found. One column returned.  
*** Total elapsed time was 1 second.
```

```
Count(*)  
-----  
497
```

Se a conexão não for bem-sucedida, verifique se o software cliente Teradata está corretamente instalado e configurado no servidor federado.

3. Efetue logon no servidor Teradata e finalize a sessão do utilitário BTEQ.

A próxima tarefa nesta seqüência de tarefas é verificar se a biblioteca Teradata está ativada para o link de tempo de execução.

Related tasks:

- “Verificando se a Biblioteca Teradata Está Ativada para Link de Tempo de Execução (AIX)” na página 130

Verificando se a Biblioteca Teradata Está Ativada para Link de Tempo de Execução (AIX)

Verificar se a biblioteca Teradata está ativada para o link de tempo de execução faz parte da maior tarefa de adicionar origens de dados Teradata a servidores federados.

Ao adicionar uma origem de dados Teradata ao servidor federado no AIX, é necessário verificar se o link de tempo de execução está ativado antes de registrar wrappers ou servidores.

Procedimento:

Para verificar se a biblioteca Teradata está ativada para link de tempo de execução:

1. Vá para o diretório no qual o arquivo libcliv2.so reside.
Por padrão, o processo de instalação coloca este arquivo no diretório /usr/lib.
2. Emita o seguinte comando do UNIX.

```
dump -H libcliv2.so | grep libtli.a
```
3. Verifique os nomes de arquivos que aparecem na tela.
Se aparecer o nome do arquivo libtli.a, isto indica que a biblioteca Teradata está ativada para o link de tempo de execução.
4. Se o nome do arquivo libtli.a não aparecer, emita os seguintes comandos do UNIX.

```
rtl_enable libcliv2.so -F libtli.a
mv libcliv2.so libcliv2.so.old
mv libcliv2.so.new libcliv2.so
chmod a+r libcliv2.so
```

Estes comandos ativam o link de tempo de execução para a biblioteca Teradata.

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper Teradata.

Related tasks:

- “Registrando o Wrapper Teradata” na página 131

Registrando o Wrapper Teradata

O registro do wrapper Teradata faz parte da maior tarefa de adicionar origens de dados Teradata a servidores federados.

Você deve emitir a instrução CREATE WRAPPER para registrar o wrapper Teradata e para identificar a biblioteca de wrapper para o servidor federado.

O wrapper Teradata, que é chamado TERADATA, está incluído no DB2 Information Integrator.

Procedimento:

Para especificar o wrapper que deseja utilizar para acessar origens de dados Teradata, emita a instrução CREATE WRAPPER.

Por exemplo:

```
CREATE WRAPPER TERADATA
```

Recomendação: Utilize o nome do wrapper padrão TERADATA quando emitir a instrução CREATE WRAPPER. Ao registrar o wrapper que utiliza o nome padrão, o servidor federado utiliza automaticamente o nome da biblioteca padrão associado a esse wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, será possível substituir o nome do wrapper padrão por um nome de sua escolha. Se você utilizar um nome que seja diferente do padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER.

Por exemplo:

```
CREATE WRAPPER my_wrapper LIBRARY 'library_name'
```

O valor *my_wrapper* é o nome do wrapper e o valor *library_name* é o nome da biblioteca para o wrapper Teradata no sistema operacional que está sendo utilizado.

Em sistemas operacionais AIX, o nome da biblioteca para o wrapper Teradata é `libdb2teradata.a`.

Em sistemas operacionais Windows, o nome da biblioteca para o wrapper Teradata é `db2teradata.dll`.

A próxima tarefa nesta seqüência de tarefas é registrar a definição de servidor para uma origem de dados Teradata.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Teradata” na página 132

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando Definições de Servidor para uma Origem de Dados Teradata

O registro de definições de servidor para uma origem de dados Teradata faz parte da maior tarefa de adicionar origens de dados Teradata a servidores federados.

No banco de dados federado, você deve definir cada servidor Teradata que deseja acessar. Primeiro, você deve localizar o nome do nó da origem de dados Teradata e, em seguida, utilizar este nome do nó ao registrar o servidor.

Procedimento:

Para registrar uma definição de servidor para uma origem de dados Teradata:

1. Localize o nome do nó.

a. Localize o arquivo `hosts`.

Em sistemas operacionais AIX, o arquivo `hosts` é `/etc/hosts`.

Em sistemas operacionais Windows, o arquivo `hosts` é `x:\WINNT\system32\drivers\etc\hosts`. *x*: é a unidade na qual o diretório `\WINNT` reside.

b. Pesquise no arquivo `hosts` o alias do servidor remoto.

Este alias começa com uma cadeia de símbolos alfabéticos e termina com o sufixo `COPn`. O valor *n* é o número do processador do aplicativo que está associado ao processador de comunicações Teradata.

- c. Localize a linha no arquivo hosts que contém este alias.
- d. Localize o primeiro campo não numérico nessa linha.

Exemplo arquivo hosts:

```
127.0.0.1      localhost

9.22.5.77     nodexyz      nodexyzC0P1      # teradata server

9.66.111.133  rtp1ib05.data.xxx.com aap
9.66.111.161 rtpscm11.data.xxx.com aaprwr
9.66.111.161 rtpscm11.data.xxx.com accessm
```

Neste exemplo, o campo **nodexyz** é o nome do nó.

2. Emita a instrução CREATE SERVER.

Por exemplo:

```
CREATE SERVER server_name TYPE TERADATA VERSION 4 WRAPPER wrapper
      OPTIONS (NODE 'node_name')
```

Você deve especificar um nome do servidor. O nome especificado deve ser exclusivo.

Você deve definir o parâmetro TYPE como *TERADATA* para todos os servidores Teradata.

O wrapper Teradata suporta todas as versões do Teradata V2R3 e do Teradata V2R4. Você pode especificar o número da versão como um dígito ou como dois dígitos com um ponto decimal. Exemplos de números de versões válidos incluem 3, 3.0, 3.5, 4, 4.0, 4.4 e outros.

Você deve especificar um nome para o wrapper. O nome especificado corresponde a um wrapper Teradata registrado com a instrução CREATE WRAPPER.

Também é necessário especificar o nome do nó no qual o servidor Teradata reside. Este nome de nó faz distinção entre maiúsculas e minúsculas.

Ao registrar uma definição de servidor Teradata, você pode especificar opções de servidor adicionais na instrução CREATE SERVER, se requerido.

Depois de registrar a definição de servidor, você pode adicionar ou eliminar opções de servidor, emitindo a instrução ALTER SERVER.

A próxima tarefa nesta seqüência de tarefas é criar o mapeamento de usuário para uma origem de dados Teradata.

Related tasks:

- “Criando o Mapeamento de Usuário para uma Origem de Dados Teradata” na página 135

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- “Instrução CREATE SERVER - Exemplos para Wrapper Teradata” na página 134

Instrução CREATE SERVER - Exemplos para Wrapper Teradata

Este tópico fornece vários exemplos que mostram como utilizar a instrução CREATE SERVER para registrar servidores para o wrapper Teradata. Este tópico inclui um exemplo completo, que mostra como criar um servidor com todos os parâmetros obrigatórios e um exemplo com opções de servidor opcionais.

Exemplo completo:

O exemplo a seguir mostra como criar uma definição de servidor para um wrapper Teradata utilizando a instrução CREATE SERVER:

```
CREATE SERVER TERASERVER TYPE TERADATA VERSION 4 WRAPPER my_wrapper
        OPTIONS (NODE 'tera_node')
```

A opção de servidor *TERASERVER* especifica o nome atribuído ao servidor Teradata. *TYPE TERADATA* especifica que você está configurando o acesso a uma origem de dados Teradata. *VERSION 4* é a versão de software do servidor Teradata que você deseja acessar. *WRAPPER my_wrapper* especifica o nome do wrapper Teradata registrado através da instrução CREATE WRAPPER. *NODE 'tera_node'* é o nome do nó no qual o servidor Teradata reside.

Exemplo de opção do servidor:

O exemplo a seguir mostra uma definição de servidor Teradata com estatísticas para o otimizador:

```
CREATE SERVER TERASERVER1 TYPE TERADATA
        VERSION 4 WRAPPER WRAPPERNAME1
        OPTIONS (NODE 'tera_node1', CPU_RATIO '2.0', IO_RATIO '3.0')
```

Neste exemplo, *TERASERVER1* é o nome do servidor Teradata, *WRAPPERNAME1* é o nome do wrapper registrado através da instrução CREATE WRAPPER e *'tera_node1'* é o nome do nó no qual o servidor Teradata reside. As opções de servidor *CPU_RATIO* e *IO_RATIO* fornecem as seguintes informações para o otimizador:

- Os recursos da CPU do servidor federado são duas vezes mais potentes que os recursos da CPU do servidor Teradata.
- Os dispositivos de E/S do servidor federado processam dados três vezes mais rápido que o dispositivos de E/S do servidor Teradata.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados Teradata” na página 132

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Criando o Mapeamento de Usuário para uma Origem de Dados Teradata

A criação do mapeamento de usuário para uma origem de dados Teradata faz parte da maior tarefa de adicionar origens de dados Teradata a servidores federados.

Ao tentar acessar um servidor Teradata, o servidor federado estabelece uma conexão com a origem de dados utilizando um ID do usuário e senha válidos para essa origem de dados. É necessário definir uma associação (mapeamento do usuário) entre cada ID do usuário do servidor federado e o ID do usuário e senha da origem de dados correspondente. 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.

Você deve criar mapeamentos de usuários para cada servidor Teradata registrado na instrução CREATE SERVER associada.

Procedimento:

Para mapear o ID do usuário federado para o ID do usuário e senha do servidor Teradata, emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR USERID SERVER TERASERVER
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

Como alternativa, você pode criar mapeamentos de usuários utilizando a janela Criar Mapeamento de Usuário do Centro de Controle do DB2.

A próxima tarefa nesta seqüência de tarefas é testar a conexão do servidor federado com o servidor Teradata.

Related tasks:

- “Testando a Conexão do Servidor Federado com o Servidor Teradata” na página 137

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Instrução CREATE USER MAPPING - Exemplos para o Wrapper Teradata” na página 136

Instrução CREATE USER MAPPING - Exemplos para o Wrapper Teradata

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE USER MAPPING para mapear um ID do usuário federado local para um ID do usuário e senha do servidor Teradata. Este tópico inclui um exemplo completo com todos os parâmetros obrigatórios e um exemplo que mostra como utilizar o registro especial USER do DB2 com a instrução CREATE USER MAPPING.

Exemplo completo:

O exemplo a seguir mostra como mapear um ID do usuário federado local (*MICHAEL*) para um ID do usuário e senha do servidor Teradata (*'mike'* e *'passxyz123'*):

```
CREATE USER MAPPING FOR MICHAEL SERVER TERASERVER
    OPTIONS (REMOTE_AUTHID 'mike', REMOTE_PASSWORD 'passxyz123')
```

A opção *MICHAEL* especifica o ID do usuário federado que você está mapeando para um ID do usuário definido no servidor Teradata. *SERVER TERASERVER* especifica o nome do servidor Teradata definido na instrução CREATE SERVER. *REMOTE_AUTHID 'mike'* é o ID do usuário no servidor Teradata para o qual você está mapeando o ID do usuário local chamado *MICHAEL*. *REMOTE_PASSWORD 'passxyz123'* é a senha associada ao valor *REMOTE_AUTHID* de *'mike'*.

Exemplo especial de registro:

O exemplo a seguir mostra uma instrução CREATE USER MAPPING que inclui o registro especial USER:

```
CREATE USER MAPPING FOR USER SERVER TERASERVER
    OPTIONS (REMOTE_AUTHID 'mike', REMOTE_PASSWORD 'passxyz123')
```

Você pode utilizar o registro especial USER 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 REMOTE_AUTHID do usuário.

Related tasks:

- “Criando o Mapeamento de Usuário para uma Origem de Dados Teradata” na página 135

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Testando a Conexão do Servidor Federado com o Servidor Teradata

O teste da conexão do servidor federado com o servidor Teradata faz parte da maior tarefa de adicionar origens de dados Teradata a servidores federados.

Você pode testar a conexão do servidor federado com o servidor Teradata utilizando a definição de servidor e o mapeamento de usuário definidos.

Procedimento:

Para testar a conexão:

1. A partir do processador da linha de comandos do DB2, abra uma sessão de passagem para emitir uma instrução SQL SELECT em uma tabela do sistema Teradata.

Por exemplo:

```
SET PASSTHRU server_name
SELECT count(*) FROM dbc.tables
SET PASSTHRU RESET
```

Se a instrução SQL SELECT retornar uma contagem, a definição de servidor e o mapeamento do usuário estarão configurados adequadamente.

2. Se a instrução SQL SELECT retornar um erro, é possível que seja necessário:
 - Verificar o servidor Teradata para certificar-se de que ele esteja configurado para conexões de entrada.
 - Verificar o mapeamento de usuário para certificar-se de que as definições para as opções REMOTE_AUTHID e REMOTE_PASSWORD sejam válidas para as conexões com o servidor Teradata. Alterar o mapeamento de usuário ou criar outro mapeamento, conforme necessário.

- Verificar o software cliente Teradata no servidor federado DB2 para certificar-se de que o software esteja instalado e configurado para conectar-se ao servidor Teradata.
- Verificar as definições das variáveis federadas do DB2 para verificar se é possível acessar o servidor Teradata. Estas variáveis incluem as variáveis de ambiente do sistema, as variáveis db2dj.ini e a variável DB2 Profile Registry (db2set).
- Verifique sua definição de servidor. Se necessário, elimine a definição de servidor e recrie-a.

Quando iniciar uma sessão de passagem para emitir instruções SQL em objetos Teradata, não será possível enviar uma instrução SQL PREPARE com um parâmetro INTO se a instrução contiver variáveis do host.

A próxima etapa nesta seqüência de tarefas é registrar pseudônimos para tabelas e exibições do Teradata.

Related concepts:

- “Definições do Servidor e Opções do Servidor” na página 10

Related tasks:

- “Registrando Pseudônimos para Tabelas e Exibições do Teradata” na página 139
- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Verificando as variáveis de ambiente de origem de dados” no *DB2 Information Integrator Installation Guide*

Related reference:

- “ALTER USER MAPPING statement” no *SQL Reference, Volume 2*

Pseudônimos Teradata em Servidores Federados

Você deve criar um pseudônimo para cada tabela e exibição do Teradata® que deseja acessar em cada servidor Teradata definido. Utilize estes pseudônimos, em vez dos nomes de objetos de origem de dados, quando consultar os servidores Teradata.

O servidor federado conecta-se à origem de dados Teradata utilizando o pseudônimo atribuído com a instrução CREATE NICKNAME. O servidor federado então consulta o catálogo da origem de dados e verifica a conexão com a origem de dados. Se a conexão não funcionar, o DB2® gerará uma mensagem de erro.

O banco de dados federado depende das estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento da consulta. Estas estatísticas são reunidas quando você cria um pseudônimo para um objeto da origem de dados.

O banco de dados federado verifica a presença do objeto na origem de dados e, em seguida, tenta reunir dados estatísticos existentes dessa origem de dados. As informações úteis ao otimizador são lidas a partir dos catálogos de origem de dados e colocadas no catálogo global no servidor federado. Como algumas ou todas as informações do catálogo da origem de dados podem ser utilizadas pelo otimizador, atualize as estatísticas na origem de dados antes de criar um pseudônimo. Atualize estas estatísticas na origem de dados utilizando um comando ou utilitário que seja equivalente ao comando **RUNSTATS** do DB2.

Você não pode enviar uma instrução SQL INSERT, UPDATE ou DELETE para um pseudônimo que se refere a uma exibição do Teradata atualizável, a menos que essa instrução SQL possa receber totalmente um pushdown para a origem de dados Teradata.

Related tasks:

- “Registrando Pseudônimos para Tabelas e Exibições do Teradata” na página 139

Related reference:

- “RUNSTATS Command” na publicação *Command Reference*
- “Instrução CREATE NICKNAME - Exemplos para Wrapper Teradata” na página 140

Registrando Pseudônimos para Tabelas e Exibições do Teradata

O registro de pseudônimos para tabelas de exibições do Teradata faz parte da maior tarefa de adicionar origens de dados Teradata a servidores federados.

Para cada servidor Teradata definido, registre um pseudônimo para cada tabela e exibição que deseja acessar.

Procedimento:

Para registrar um pseudônimo, emita a instrução CREATE NICKNAME.

Por exemplo:

```
CREATE NICKNAME TERANICKNAME FOR TERASERVER."remote_schema"."remote.table"
```

Recomendação: Como o banco de dados federado utiliza estatísticas do catálogo para objetos com pseudônimo para otimizar o processamento de consultas, atualize as estatísticas na origem de dados Teradata antes de registrar um pseudônimo. Você pode utilizar um comando ou utilitário que seja equivalente ao comando **RUNSTATS** do DB2.

Os pseudônimos podem ter até 128 caracteres.

Você pode especificar a opção de coluna `NUMERIC_STRING` ao emitir a instrução `CREATE NICKNAME`. Também pode especificar esta opção de coluna utilizando a instrução `ALTER NICKNAME`.

Related concepts:

- “Pseudônimos Teradata em Servidores Federados” na página 138

Related reference:

- “`RUNSTATS` Command” na publicação *Command Reference*
- “`CREATE NICKNAME` statement” no *SQL Reference, Volume 2*
- “Instrução `CREATE NICKNAME` - Exemplos para Wrapper Teradata” na página 140

Instrução `CREATE NICKNAME` - Exemplos para Wrapper Teradata

Este tópico fornece um exemplo que mostra como utilizar a instrução `CREATE NICKNAME` para registrar um pseudônimo para uma tabela ou exibição do Teradata que você deseja acessar.

Este exemplo mostra como especificar um objeto remoto para o servidor Teradata com o qual o pseudônimo é atribuído:

```
CREATE NICKNAME TERASALES FOR TERASERVER."salesdata"."europe"
```

`TERASALES` é o pseudônimo exclusivo que você atribui à tabela ou exibição do Teradata. Um pseudônimo é um nome com duas partes: o esquema e o pseudônimo real. Se você omitir o esquema quando criar o pseudônimo, o DB2 criará o pseudônimo utilizando seu ID de autenticação como o esquema.

`TERASERVER."salesdata"."europe"` especifica um identificador de três partes para o objeto remoto:

- `TERASERVER` é o nome atribuído ao servidor do banco de dados Teradata na instrução `CREATE SERVER`.
- `salesdata` é o nome do esquema remoto ao qual a tabela ou exibição pertence.
- `europe` é o nome da tabela ou exibição remota que você deseja acessar.

Related concepts:

- “Pseudônimos Teradata em Servidores Federados” na página 138

Related tasks:

- “Registrando Pseudônimos para Tabelas e Exibições do Teradata” na página 139

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*

Ajustando e Resolvendo Problemas de Configuração para Origens de Dados Teradata

Depois de fazer a configuração para origens de dados Teradata, você pode alterar a configuração para aprimorar o desempenho e para eliminar possíveis erros.

Aprimorando o Acesso ao Servidor Teradata Definindo a Variável DB2_DJ_COMM

Talvez você queira definir a variável de registro do perfil DB2_DJ_COMM DB2 para aprimorar o desempenho, se achar que demora muito tempo para acessar o servidor Teradata. Quando definir a variável DB2_DJ_COMM, o servidor federado carregará o wrapper durante a inicialização em vez de quando você tenta acessar a origem de dados.

Procedimento:

Para definir a variável DB2_DJ_COMM:

1. Defina a variável DB2_DJ_COMM para o wrapper da biblioteca correspondente ao wrapper especificado. Utilize um dos seguintes comandos para definir a variável de registro do perfil DB2_DJ_COMM DB2:

No AIX:

```
db2set DB2_DJ_COMM='libdb2teradata.a','libdb2teradataF.a','libdb2teradataU.a'
```

No Windows:

```
db2set DB2_DJ_COMM='db2teradata.d11','db2teradataF.d11','db2teradataU.d11'
```

2. Emita os comandos a seguir para reciclar a instância do DB2:

```
db2stop  
db2start
```

Ao reciclar a instância do DB2, assegure que ela aceite alterações de variáveis feitas.

Ajustando e Desativando o Registro de Acesso de Teradata

O produto Teradata fornece um registro de registro de acesso que gera entradas do log quando o Teradata verifica os privilégios de segurança específicos de vários usuários em um ou mais bancos de dados. Embora o registro de acesso forneça informações de segurança consideráveis e significativas, este recurso aumenta de forma significativa o uso do processador e pode reduzir o desempenho do sistema.

Se precisar aprimorar o desempenho do sistema, avalie as regras de verificação de privilégios definidas para o registro de acesso. Em seguida, finalize as regras desnecessárias, definindo instruções END LOGGING.

Para melhor desempenho, desative todo o registro de acesso. Elimine a macro DBC.AccLogRules do Teradata e, em seguida, force uma redefinição do TPA (trusted parallel application) a parar totalmente o registro de acesso.

Consulte a documentação do Teradata para obter mais informações.

Ativando o Link de Tempo de Execução para libcliv2.so (AIX)

Se você executar o arquivo `djxlinkTeradata.sh` para vincular-se à biblioteca compartilhada do Teradata chamada `libcliv2.so`, poderá receber uma mensagem de erro quando emitir uma instrução CREATE NICKNAME.

Um exemplo de uma mensagem de erro que pode ser recebida é:

```
DB21034E O comando foi processado como uma instrução SQL porque
não era um comando válido do Processador da Linha de Comandos.
Durante o processamento de SQL, ele retornou:
SQL30081N Foi detectado um erro de comunicação. Protocolo de comunicação
utilizado: "TCP/IP". API de comunicação utilizada: "SOCKETS". Localização
na qual o erro foi detectado: "9.112.26.28". Função de comunicação que detectou
o erro: "recv". Códigos de erro específicos de protocolo: "*", "*", "0".
SQLSTATE=08001
```

Se você receber uma mensagem de erro, verifique o diretório `/sqllib/db2dump` para qualquer arquivo trap. Os nomes de arquivos trap começam com a letra `t` e terminam com um sufixo `000`. Por exemplo:

```
t123456.000
```

Verifique as informações de rastreamento no arquivo trap para qualquer referência de função `OsCall` que indique que a função `OsCall` fez com que o servidor federado parasse.

O exemplo a seguir mostra informações de rastreamento com uma referência de função `OsCall` que pode ser localizada em um arquivo trap:

```
*** Start stack traceback ***
```

```
0x239690E0 OsCall + 0x28C
```

```
0x23973FB0 mtdpassn + 0x8A4
0x239795A4 mtdp + 0x208
0x2395A928 MTDPIO + 0x28C
0x239609C4 CLICON + 0xD50
0x23962350 DBCHCL + 0xC4
```

Se você localizar uma referência de função `OsCall` em um dos arquivos `trap`, emita os seguintes comandos UNIX:

```
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
```

Estes comandos ativam o link de tempo de execução para a biblioteca compartilhada de `libcliv2.so`.

Related tasks:

- “Adicionando Origens de Dados Teradata a Servidores Federados” na página 127
- “Verificando se a Biblioteca Teradata Está Ativada para Link de Tempo de Execução (AIX)” na página 130

Related reference:

- “db2set - DB2 Profile Registry Command” em *Command Reference*

Capítulo 11. Configurando o Acesso às Origens de Dados do OLE DB

Este capítulo explica como configurar seu servidor federado para acessar dados armazenados em origens de dados do OLE DB.

Este capítulo lista as tarefas que precisam ser executadas e mostra exemplos das instruções SQL que precisam ser emitidas ao fazer a configuração das origens de dados do OLE DB.

Adicionando Origens de Dados OLE DB a Servidores Federados

Microsoft OLE DB é um conjunto de interfaces OLE/COM que fornece aplicativos com acesso uniforme a dados armazenados em diversas origens de informações. A arquitetura DBMS do componente OLE DB define os consumidores OLE DB e os provedores OLE DB. Um consumidor OLE DB é qualquer sistema ou aplicativo que consome interfaces OLE DB. Um provedor OLE DB é um componente que expõe interfaces OLE DB.

O wrapper OLE DB permite acessar provedores OLE DB que são compatíveis com o Microsoft OLE DB 2.0 ou posterior.

O wrapper OLE DB é suportado em servidores federados DB2 que são executados no sistema operacional Windows NT ou Windows 2000.

Utilize o wrapper OLE DB para criar funções de tabelas. Não é possível utilizar o wrapper para criar pseudônimos em tabelas e exibições de origem de dados.

A configuração do servidor federado para acessar origens de dados OLE DB envolve o fornecimento de informações ao servidor federado sobre os provedores OLE DB.

Você pode configurar o acesso a origens de dados OLE DB através do Centro de Comandos do DB2 ou do processador da linha de comandos do DB2.

Depois de 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.

Pré-requisitos:

- Acesse o Centro de Comandos do DB2 ou o processador da linha de comandos do DB2.
- Um servidor federado e um banco de dados que são configurados para acessar origens de dados OLE DB.
- O driver OLE DB 2.0 ou posterior e um provedor OLE DB que estão instalados e configurados no servidor federado.

Restrição:

O wrapper OLE DB é utilizado apenas para ajudar no registro de tabelas externas OLE DB definidas pelo usuário. Diferente de outros wrappers, o wrapper OLE DB não utiliza pseudônimos para acessar dados armazenados em origens de dados.

Procedimento:

Para adicionar uma origem de dados OLE DB a um servidor federado:

1. Registre o wrapper.
2. Registre a definição de servidor.
3. Crie os mapeamentos do usuário.

Related concepts:

- “Trilha Rápida para Configurar Origens de Dados” na página 1

Related tasks:

- “Verificando a Instalação do Servidor Federado” no *DB2 Information Integrator Installation Guide*
- “Registrando o Wrapper OLE DB” na página 146

Registrando o Wrapper OLE DB

O registro do wrapper OLE DB faz parte da maior tarefa de adicionar origens de dados OLE a servidores federados.

Você deve emitir a instrução CREATE WRAPPER para registrar um wrapper que acessará origens de dados OLE DB.

Procedimento:

Para registrar o wrapper, emita a instrução CREATE WRAPPER.

Por exemplo:

```
CREATE WRAPPER OLEDB
```

Recomendação: Utilize o nome do wrapper padrão chamado OLEDB quando emitir a instrução CREATE WRAPPER. Ao registrar o wrapper que utiliza o nome padrão, o servidor federado utiliza automaticamente o nome da biblioteca padrão associado a esse wrapper. Se o nome do wrapper entrar em conflito com um nome de wrapper existente no banco de dados federado, será possível substituir o nome do wrapper padrão por um nome de sua escolha. Se você utilizar um nome que seja diferente do padrão, deverá incluir o parâmetro LIBRARY na instrução CREATE WRAPPER. Por exemplo:

```
CREATE WRAPPER mywrapper LIBRARY 'db2oledb.dll'
```

A próxima tarefa nesta seqüência de tarefas é registrar definições de servidor para uma origem de dados OLE DB.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados OLE DB” na página 147

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando Definições de Servidor para uma Origem de Dados OLE DB

O registro de definições de servidor para uma origem de dados OLE DB faz parte da maior tarefa de adicionar origens de dados OLE DB a servidores federados.

No banco de dados federado, você deve definir cada servidor da origem de dados OLE DB que deseja acessar.

Procedimento:

Para registrar uma definição de servidor para uma origem de dados OLE DB:

Emita a instrução CREATE SERVER.

Por exemplo:

```
CREATE SERVER server_name WRAPPER OLEDB  
    OPTIONS (CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;  
                Data Source=c:\msdasdk\bin\oledb\nwind.mdb')
```

A próxima tarefa nesta seqüência de tarefas é criar um mapeamento de usuário para uma origem de dados OLE DB.

Related tasks:

- “Criando um Mapeamento de Usuário para uma Origem de Dados OLE DB” na página 149

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- “Instrução CREATE SERVER - Exemplos para o Wrapper OLE DB” na página 148

Instrução CREATE SERVER - Exemplos para o Wrapper OLE DB

Este tópico fornece um exemplo que mostra como utilizar a instrução CREATE SERVER para registrar servidores para o wrapper OLE DB.

O exemplo a seguir mostra uma instrução CREATE SERVER:

```
CREATE SERVER Nwind WRAPPER OLEDB
OPTIONS (CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;
                    Data Source=c:\msdasdk\bin\oledb\nwind.mdb',
        COLLATING_SEQUENCE 'Y')
```

Nwind Um nome atribuído à origem de dados OLE DB. Esse nome deve ser exclusivo. Não são permitidos nomes de servidores duplicados.

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 propriedades de inicialização que são necessárias para conectar-se a uma origem de dados.

A cadeia contém uma série de pares de palavra-chave/valor que são separados por ponto e vírgula. O sinal de igual (=) separa cada palavra-chave e seu valor. Palavras-chave são descrições das propriedades de inicialização do OLE DB (conjunto de propriedades DBPROPSET_DBINT) ou de palavras-chave específicas do provedor.

Para obter a sintaxe completa e a semântica da opção CONNECTSTRING, consulte o *Microsoft OLE DB 2.0 Programmer's Reference and Data Access SDK*, Microsoft Press, 1998.

COLLATING_SEQUENCE 'Y'

Especifica se a origem de dados utiliza a mesma seqüência de intercalação que o DB2 para UNIX e Windows.

Os valores válidos são 'Y' (a origem de dados utiliza a seqüência de intercalação do DB2 para UNIX e Windows) e 'N' (a origem de dados utiliza uma seqüência de intercalação diferente da seqüência de intercalação do DB2 para UNIX e Windows). O valor padrão é 'N'.

Related tasks:

- “Registrando Definições de Servidor para uma Origem de Dados OLE DB” na página 147

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Criando um Mapeamento de Usuário para uma Origem de Dados OLE DB

A criação de um mapeamento de usuário para uma origem de dados OLE faz parte da maior tarefa de adicionar origens de dados OLE a servidores federados.

Ao tentar acessar uma origem de dados OLE, o servidor federado estabelece uma conexão com a origem de dados utilizando um ID do usuário e senha válidos para essa origem de dados. É necessário definir uma associação (mapeamento do usuário) entre cada ID do usuário do servidor federado e o ID do usuário e senha da origem de dados correspondente. Crie um mapeamento de usuário para cada ID do usuário que acessará o sistema federado para enviar pedidos distribuídos.

Procedimento:

Para mapear um ID do usuário local para o ID do usuário e senha da origem de dados OLE, emita uma instrução CREATE USER MAPPING.

Por exemplo:

```
CREATE USER MAPPING FOR userid SERVER server_name
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Instrução CREATE USER MAPPING - Exemplos para o Wrapper OLE DB” na página 149

Instrução CREATE USER MAPPING - Exemplos para o Wrapper OLE DB

Este tópico fornece exemplos que mostram como utilizar a instrução CREATE USER MAPPING para mapear um ID do usuário local para um ID do usuário e senha da origem de dados OLE. Este tópico inclui um exemplo completo com os parâmetros requeridos e um exemplo que mostra como utilizar o registro especial do DB2 USER com a instrução CREATE USER MAPPING.

Exemplo completo:

O exemplo a seguir mostra como mapear um ID do usuário local para um ID do usuário e senha da origem de dados OLE:

```
CREATE USER MAPPING FOR laura SERVER Nwind  
    OPTIONS (REMOTE_AUTHID 'lulu', REMOTE_PASSWORD 'raiders')
```

laura O ID do usuário local que você está mapeando para um ID do usuário definido na origem de dados OLE DB.

SERVER *Nwind*

O nome do servidor OLE DB definido na instrução CREATE SERVER.

REMOTE_AUTHID '*lulu*'

O ID do usuário no servidor OLE DB para o qual você está mapeando *laura*. Este valor faz distinção entre letras maiúsculas e minúsculas.

REMOTE_PASSWORD '*raiders*'

A senha associada a '*lulu*'. Este valor faz distinção entre letras maiúsculas e minúsculas.

Exemplo especial de registro:

O exemplo a seguir mostra uma instrução CREATE USER MAPPING que inclui o registro especial USER:

```
CREATE USER MAPPING FOR USER SERVER Nwind  
    OPTIONS (REMOTE_AUTHID 'lulu', REMOTE_PASSWORD 'raiders')
```

Você pode utilizar o registro especial USER 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 REMOTE_AUTHID do usuário.

Related tasks:

- “Criando um Mapeamento de Usuário para uma Origem de Dados OLE DB” na página 149

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Capítulo 12. Configurando o Acesso às Origens de Dados de Arquivos com Estrutura de Tabela

Este capítulo explica o que são os arquivos com estrutura de tabela, como incluí-los como origens de dados em seu sistema federado e lista as mensagens de erro associadas com o wrapper de arquivos com estrutura de tabela.

O que São Arquivos com Estrutura de Tabela?

Um arquivo estruturado de tabela apresenta uma estrutura regular que consiste numa série de registros, em que cada registro contém o mesmo número de campos, separados por um delimitador arbitrário. Os valores nulos são representados por dois delimitadores próximos.

O exemplo a seguir mostra o conteúdo de um arquivo chamado DRUGDATA1.TXT. Ele contém três registros, cada qual com três campos, separados por vírgula:

```
234,DrugnameA,Manufacturer1  
332,DrugnameB,Manufacturer2  
333,DrugnameC,Manufacturer2
```

O primeiro campo é o número da ID exclusiva do medicamento. O segundo campo é o nome do medicamento. O terceiro campo é o nome do fabricante que produz o medicamento.

Related concepts:

- “Tipos de Arquivos com Estrutura de Tabela” na página 152
- “Como o DB2 Information Integrator Funciona com Arquivos com Estrutura de Tabela” na página 152
- “O que É Documentum?” na página 165
- “O que É Excel?” na página 203
- “O que É BLAST?” na página 219
- “O que É XML?” na página 247

Related tasks:

- “Incluindo Arquivos com Estrutura de Tabela em um Sistema Federado” na página 154

Tipos de Arquivos com Estrutura de Tabela

Arquivo estruturado de tabela poderá ser ordenado ou não-ordenado.

Arquivos Ordenados

DRUGDATA1.TXT contém registros ordenados. O arquivo é ordenado pelo primeiro campo, o número da ID exclusivo da droga. Este campo é a chave primária porque é exclusivo para cada medicamento. Arquivos ordenados devem estar em ordem crescente.

```
234,DrugnameA,Manufacturer1
332,DrugnameB,Manufacturer2
333,DrugnameC,Manufacturer2
```

Arquivos Não-ordenados

DRUGDATA2.TXT contém registros não-ordenados. Não existe seqüência na forma que os registros estão listados no arquivo.

```
332,DrugnameB,Manufacturer2
234,DrugnameA,Manufacturer1
333,DrugnameC,Manufacturer2
```

O wrapper pode pesquisar arquivos de dados ordenados com mais eficiência que arquivos não-ordenados.

Related concepts:

- “O que São Arquivos com Estrutura de Tabela?” na página 151
- “Como o DB2 Information Integrator Funciona com Arquivos com Estrutura de Tabela” na página 152

Related tasks:

- “Incluindo Arquivos com Estrutura de Tabela em um Sistema Federado” na página 154

Como o DB2 Information Integrator Funciona com Arquivos com Estrutura de Tabela

Utilizando um módulo chamado wrapper, o DB2[®] Information Integrator pode processar instruções SQL que consultam dados em um arquivo estruturado de tabela como se estivessem contidos em uma tabela ou exibição ordinária relacional. Isto permite que os dados em um arquivo estruturado de tabela sejam unidos a dados relacionais ou dados em algum outro arquivo estruturado de tabela. Este processo é ilustrado na Figura 1 na página 153.

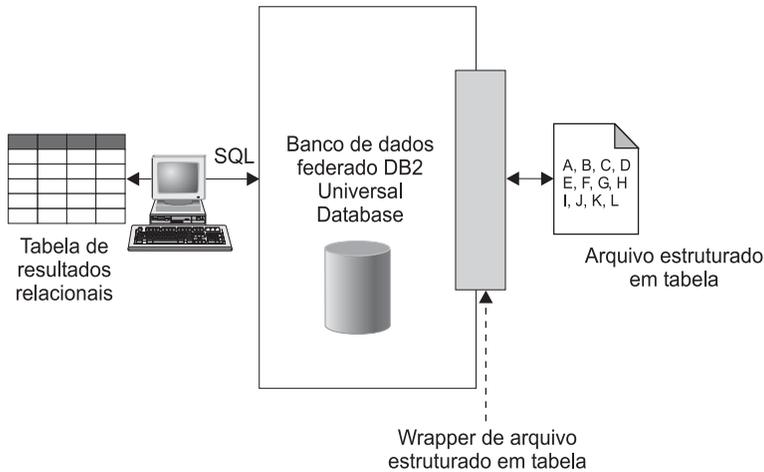


Figura 1. Como o wrapper do arquivo estruturado de tabela funciona

Por exemplo, suponha que o arquivo DRUGDATA2.TXT do arquivo estruturado de tabela esteja localizado no computador em seu laboratório. A tentativa de consultar estes dados e compará-los com outras tabelas de outras origens de dados que você usa pode ser tediosa.

Depois de registrar o DRUGDATA2.TXT no DB2 Information Integrator, o arquivo se comporta como se fosse uma origem de dados relacional. Você pode agora consultar o arquivo junto com outras origens de dados relacionais e não-relacionais e analisar os dados em conjunto.

Por exemplo, é possível executar a seguinte consulta:

```
SELECT * FROM DRUGDATA2 ORDER BY DCODE
```

Esta consulta produz os seguintes resultados:

| Dcode | Medicamento | Fabricante |
|-------|-------------|---------------|
| 234 | DrugnameA | Manufacturer1 |
| 332 | DrugnameB | Manufacturer2 |
| 333 | DrugnameC | Manufacturer2 |

Related concepts:

- “O que São Arquivos com Estrutura de Tabela?” na página 151
- “Tipos de Arquivos com Estrutura de Tabela” na página 152

Related tasks:

- “Incluindo Arquivos com Estrutura de Tabela em um Sistema Federado” na página 154

Incluindo Arquivos com Estrutura de Tabela em um Sistema Federado

Restrições:

Você não pode utilizar sessões de passagem para acessar uma origem de dados de arquivo estruturado de tabela.

Procedimento:

Para incluir uma origem de dados para um arquivo estruturado de tabela em um servidor federado:

1. Registre o wrapper utilizando o comando CREATE WRAPPER. Registre o wrapper utilizando o comando CREATE WRAPPER.
2. Opcional: Defina a variável de ambiente DB2_DJ_COMM para melhorar o desempenho da consulta.
3. Registre o servidor utilizando o comando CREATE SERVER. Registre o servidor utilizando o comando CREATE SERVER.
4. Registre pseudônimos utilizando o comando CREATE NICKNAME para todos os arquivo estruturado de tabelas. Registre pseudônimos utilizando o comando CREATE NICKNAME para todos os arquivo estruturado de tabelas.

Os comandos podem ser executados a partir do Processador da Linha de Comandos do DB2.

Related tasks:

- “Registrando o Wrapper do Arquivo Estruturado de Tabela” na página 155
- “Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper de Arquivo Estruturado de Tabela” na página 156
- “Registrando o Servidor para Arquivos com Estrutura de Tabela” na página 156
- “Registrando Pseudônimos para Arquivos com Estrutura de Tabela” na página 157

Registrando o Wrapper do Arquivo Estruturado de Tabela

O registro do wrapper de arquivo estruturada de tabela faz parte da maior tarefa de adicionar arquivos estruturados de tabela a um sistema federado. Você deve registrar o wrapper para acessar uma origem de dados. Wrappers são mecanismos que os servidores federados utilizam para se comunicar com as origens de dados e recuperar dados delas. Os wrappers são instalados no sistema como arquivos de biblioteca.

Procedimento:

Para registrar o wrapper, utilize a instrução `CREATE WRAPPER` para especificar qual wrapper será utilizado para acessar o arquivo estruturado de tabela.

Por exemplo, para registrar um wrapper no AIX, execute a seguinte instrução:

```
CREATE WRAPPER laboratory_flat_files LIBRARY 'libdb21sfile.a'  
  OPTIONS(DB2_FENCED 'N');
```

Neste exemplo, `laboratory_flat_files` é o nome escolhido para o wrapper. Esse nome deve ser exclusivo no banco de dados no qual o wrapper está sendo registrado. O nome de biblioteca requerido para o wrapper do arquivo estruturado de tabela no AIX é `libdb21sfile.a`.

O nome de biblioteca é instalado como `libdb21sfile.a` por padrão, mas pode ter sido personalizado durante a instalação. Verifique com o administrador do sistema o nome correto.

Para obter uma tabela de nomes de arquivos de biblioteca padrão para o wrapper do arquivo estruturado de tabela por plataforma suportada, consulte a seção de tarefas relacionadas.

A próxima tarefa nesta seqüência de tarefas é definir a variável de ambiente `DB2_DJ_COMM` para o wrapper do arquivo estruturada de tabela.

Related tasks:

- “Definindo a Variável de Perfil `DB2_DJ_COMM` do DB2 para o Wrapper de Arquivo Estruturado de Tabela” na página 156
- “Depois de Instalar Wrappers Não Relacionais” no *DB2 Information Integrator Installation Guide*

Related reference:

- “`CREATE WRAPPER` statement” no *SQL Reference, Volume 2*

Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper de Arquivo Estruturado de Tabela

A definição da variável de perfil DB2_DJ_COMM do DB2 para o wrapper do arquivo estruturado de tabela faz parte da maior tarefa de adicionar arquivos com estrutura de tabela a um sistema federado. Para aprimorar o desempenho quando arquivo estruturado de tabelas forem acessados, defina a variável de perfil DB2_DJ_COMM do DB2. Esta variável determina se o servidor federado carregará o wrapper na inicialização.

Procedimento:

Para definir a variável de perfil DB2_DJ_COMM do DB2, envie o comando `db2set` com a biblioteca do wrapper que corresponde ao wrapper especificado na instrução `CREATE WRAPPER` associada.

Por exemplo:

```
db2set DB2_DJ_COMM='libdb2lsfile.a'
```

Verifique se não existem espaços em nenhum lado do sinal de igual (=).

Existe overhead associado ao carregamento das bibliotecas de wrapper durante a inicialização do banco de dados. Para evitar este overhead, especifique apenas as bibliotecas que você pretende acessar.

A próxima tarefa nesta seqüência de tarefas é registrar o servidor para arquivos com estrutura de tabela.

Related tasks:

- “Registrando o Servidor para Arquivos com Estrutura de Tabela” na página 156

Registrando o Servidor para Arquivos com Estrutura de Tabela

O registro do servidor para arquivos com estrutura de tabela faz parte da maior tarefa de adicionar arquivos com estrutura de tabela a um sistema federado. Depois de registrar o wrapper, você deve registrar um servidor correspondente.

Procedimento:

Para registrar o servidor do arquivo estruturado de tabela no sistema federado, utilize a instrução `CREATE SERVER`. Por exemplo:

```
CREATE SERVER biochem_lab WRAPPER laboratory_flat_files
```

Neste exemplo, `biochem_lab` é o nome atribuído ao servidor arquivo estruturado de tabela. O nome deve ser exclusivo para o banco de dados em que o servidor está sendo registrado.

A próxima tarefa nesta sequência de tarefas é registrar pseudônimos para arquivos com estrutura de tabela.

Related tasks:

- “Registrando Pseudônimos para Arquivos com Estrutura de Tabela” na página 157

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Registrando Pseudônimos para Arquivos com Estrutura de Tabela

O registro de pseudônimos para arquivos com estrutura de tabela faz parte da maior tarefa de adicionar arquivos com estrutura de tabela a um sistema federado. Depois de registrar o servidor, você deve registrar um pseudônimo correspondente. Pseudônimos são utilizados quando você faz referência a uma origem de dados de um arquivo estruturado de tabela numa consulta.

Os pseudônimos são associados ao arquivo estruturado de tabela de uma de duas formas:

- em um modo fixo utilizando a opção de pseudônimo `FILE_PATH`. Quando esta opção é utilizada, o pseudônimo representa dados de um arquivo estruturado de tabela específico.
- com um nome do arquivo especificado no momento da consulta utilizando a opção de coluna do pseudônimo `DOCUMENT`. Quando esta opção é utilizada, o pseudônimo pode ser utilizado para representar dados de qualquer arquivo estruturada de tabela cujo esquema corresponda à definição de pseudônimo.

Restrições:

Se um campo não-numérico for muito longo para seu tipo de coluna, os dados excedentes serão truncados. Se um campo decimal no arquivo tiver mais dígitos após o caractere do radical que o permitido pelo parâmetro de escala de seu tipo de coluna, os dados excedentes serão truncados. O caractere base é determinado pelo item `RADIXCHAR` da categoria `LC_NUMERIC` do `NLS (National Language Support)`.

O comprimento máximo da linha é 32768.

Arquivos contendo caracteres de múltiplo byte não são suportados.

Procedimento:

Para registrar um pseudônimo, utilize a instrução CREATE NICKNAME para cada arquivo estruturado de tabela que quiser acessar.

Não há mais tarefas nesta seqüência de tarefas.

Related tasks:

- “Depois de Instalar Wrappers Não Relacionais” no *DB2 Information Integrator Installation Guide*

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper do Arquivo Estruturado de Tabela” na página 370
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper do Arquivo Estruturado Tabela” na página 158

Instrução CREATE NICKNAME - Exemplos para o Wrapper do Arquivo Estruturado Tabela

Este tópico fornece um exemplo completo de utilização de uma instrução CREATE NICKNAME para registrar pseudônimos para o wrapper do arquivo estruturado de tabela. Ele também inclui exemplos para opções específicas.

Exemplo completo:

O exemplo a seguir mostra uma instrução CREATE NICKNAME para arquivo estruturado de 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')
```

Exemplos de opções KEY COLUMN:

Estes exemplos mostram que a coluna designada como a chave é designada como não anulável, adicionando a opção NOT NULL à sua definição na instrução do pseudônimo:

```
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')
```

Esta opção faz distinção entre maiúsculas e minúsculas. Entretanto, o DB2 dobra nomes de coluna para letras maiúsculas a menos que a coluna seja definida com aspas duplas. O exemplo a seguir não funcionará corretamente porque a coluna empno será dobrada para letras maiúsculas pelo DB2 e a coluna chave empno será enviada em letras minúsculas. Assim, a coluna designada como chave não será encontrada.

```
CREATE NICKNAME depart (
  empno char(6) NOT NULL)
FOR SERVER DATASTORE
  OPTIONS(FILE_PATH'data.txt', SORTED 'Y', KEY_COLUMN 'empno');
```

Related tasks:

- “Registrando Pseudônimos para Arquivos com Estrutura de Tabela” na página 157

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper do Arquivo Estruturado de Tabela” na página 370

Modelo de Controle de Acesso de Arquivo para o Wrapper do Arquivo Estruturado de Tabela

O sistema de gerenciamento de bancos de dados acessará os arquivos com estrutura de tabela com a autoridade do proprietário da instância do DB2. O wrapper pode acessar somente arquivos que podem ser lidos por esta ID de usuário (ou ID de grupo). A ID de autorização do aplicativo (ID que estabelece a conexão ao banco de dados federado) não é relevante.

No DB2 Universal Database Enterprise Server Edition, todo arquivo estruturado de tabela para o qual um pseudônimo foi criado, deve estar acessível com o mesmo nome do caminho de cada nó. O arquivo não precisa ser um nó do DB2 Universal Database, desde que possa ser acessado a partir de qualquer nó com um caminho comum.

Para acessar um arquivo estruturado de tabela em uma unidade mapeada, se a rede tiver um domínio do Windows NT ou Windows 2000 configurado, a conta de logon do serviço DB2 deverá ser uma conta do domínio que tem acesso à pasta compartilhada na qual os arquivos de dados residem.

Para acessar um arquivo estruturado de tabela em uma unidade mapeada, se a rede não tiver um domínio do Windows NT ou Windows 2000 e seu usuário

efetuar logon localmente em cada estação de trabalho, a conta de logon do serviço DB2 deverá ter o mesmo nome de usuário e senha que um usuário válido na máquina que compartilha essa pasta. Esse usuário deve estar na lista de permissões para a pasta compartilhada com, pelo menos, acesso de leitura.

Related reference:

- “Controle de Acesso para o Wrapper do Documentum” na página 194
- “Modelo de Controle de Acesso de Arquivo para o Wrapper do Excel” na página 210

Dicas e Considerações de Otimização para o Wrapper do Arquivo Estruturado de Tabela

- O sistema pode pesquisar arquivos de dados ordenados com mais eficiência que arquivos não-ordenados.
- Em arquivos ordenados, é possível melhorar o desempenho especificando-se um valor ou intervalo para a coluna-chave quando uma consulta é enviada.
- A estatística para pseudônimos de arquivo estruturado de tabela deve ser atualizada manualmente através da atualização das exibições SYSSTAT e SYSCAT.

Related reference:

- “Dicas de Otimização para o Wrapper do BLAST” na página 243

Mensagens para o Wrapper do Arquivo Estruturado de Tabela

Esta seção lista e descreve mensagens que podem ser encontradas enquanto se trabalha com o wrapper do arquivo estruturado de tabela.

Tabela 22. Mensagens emitidas pelo wrapper para o arquivo estruturado de tabela

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL0405N | O literal numérico "<literal>" não é válido porque seu valor está fora da faixa. | Uma coluna o arquivo de dados ou um valor de predicado em uma instrução SQL, contém um valor que está fora do intervalo possível para esse tipo de dados. Corrija o arquivo de dados ou redefina a coluna com um tipo mais adequado. |

Tabela 22. Mensagens emitidas pelo wrapper para o arquivo estruturado de tabela (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|--|
| SQL0408N | Um valor não é compatível com o tipo de dados do destino da atribuição. O nome de destino é "<nome_da_coluna>". | Uma coluna no arquivo de dados contém caracteres que são inválidos para esse tipo de dados. Corrija o arquivo de dados ou redefina a coluna com um tipo mais adequado. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "O caminho da origem de dados é NULL".) | Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Falha na recuperação da Coluna-Chave".) | Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Falha de STAT na origem de dados. ERRNO = <número_do_erro>".) | Certifique-se de ter as permissões de diretório apropriadas. Certifique-se de que o arquivo exista. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Não encontradas informações da coluna".) | Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Operador não-suportado".) | Contate o Suporte de Software da IBM. |

Tabela 22. Mensagens emitidas pelo wrapper para o arquivo estruturado de tabela (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|---|
| SQL1816N | O wrapper " <code><nome_do_wrapper></code> " não pode ser usado para acessar o "tipo" de origem de dados (" <code><tipo></code> " "") que você está tentando definir para o banco de dados federado. | O tipo de servidor é inválido. Nenhum tipo de servidor deve ser especificado na instrução CREATE SERVER. Remova o valor e a palavra-chave TYPE e execute-a novamente. |
| SQL1822N | Código de erro imprevisto "ERRNO = <code><número_do_erro></code> " recebido a partir da origem de dados " <code><nome_do_servidor></code> ". O texto e os tokens associados são "Impossível ler o arquivo". | Verifique o valor do número do erro. Verifique se o arquivo pode ser lido pelo proprietário da instância do DB2. Em seguida, execute novamente o comando SQL. |
| SQL1822N | Código de erro imprevisto "Erros de Dados" recebido a partir da origem de dados " <code><nome_do_servidor></code> ". O texto e os tokens associados são "A origem de dados é um arquivo não padrão". | A origem de dados é um diretório, soquete ou FIFO. Somente arquivos padrão podem ser acessados como origem de dados. Altere a opção FILE_PATH para apontar a um arquivo válido e emita novamente o comando SQL. |
| SQL1822N | Código de erro imprevisto "ERRNO = <code><número_do_erro></code> " recebido a partir da origem de dados " <code><nome_do_servidor></code> ". O texto e os tokens associados são "Erro de abertura do arquivo". | O wrapper não conseguiu abrir o arquivo. Verifique o número do erro para determinar a causa de ocorrência do erro. Corrija o problema com a origem de dados e emita novamente o comando SQL. |
| SQL1822N | Código de erro imprevisto "Erros de Dados" recebido a partir da origem de dados " <code><nome_do_servidor></code> ". O texto e os tokens associados são "Falta a coluna-chave". | Estava faltando o campo-chave de um registro recuperado da origem de dados. A coluna-chave não deve ser nula. Corrija os dados ou registre o arquivo com um pseudônimo não-ordenado. |

Tabela 22. Mensagens emitidas pelo wrapper para o arquivo estruturado de tabela (continuação)

| Código de Erro | Mensagem | Explicação |
|-----------------------|---|---|
| SQL1822N | Código de erro imprevisto "Erros de Dados" recebido a partir da origem de dados "<nome_do_servidor>". O texto e os tokens associados são "Arquivo não-ordenado". | O arquivo não foi ordenado na coluna-chave. Proceda de uma das seguintes formas: altere a opção KEY_COLUMN para apontar para a coluna correta; reorganize o arquivo de dados ou registre o pseudônimo como não-ordenado. |
| SQL1822N | Código de erro imprevisto "Erros de Dados" recebido a partir da origem de dados "<nome_do_servidor>". O texto e os tokens associados são "A chave excede o tamanho da definição". | O campo coluna-chave lido da origem de dados era maior que a definição de colunas do DB2, o que poderia fazer com que as rotinas de pesquisa do wrapper funcionassem de forma incorreta. Corrija os dados ou a definição do pseudônimo e registre o pseudônimo novamente. |
| SQL1822N | Código de erro imprevisto "Erros de Dados" recebido a partir da origem de dados "<nome_do_servidor>". O texto e os tokens associados são "A linha no arquivo de dados excede 32 K". | Uma linha no arquivo de dados excede o comprimento da linha máximo permitido pelo wrapper. O comprimento da linha não pode ser maior que 32768. Diminua o comprimento da linha no arquivo de dados. |
| SQL1823N | Não existe mapeamento de tipo de dados para o tipo de dados "<tipo_de_dados>" do servidor "<nome_do_servidor>". | O pseudônimo foi definido com um tipo de dados não suportado. Redefina o pseudônimo usando somente tipos de dados suportados. |
| SQL1881N | "<nome_da_opção>" não é uma opção válida do "<componente>" para "<nome_do_objeto>". | O valor listado não é uma opção válida para o objeto relacionado. Remova ou altere a opção inválida e envie a instrução SQL novamente. |
| SQL1882N | "COLUMN_DELIMITER" da opção "Pseudônimo" não pode ser definido em "<delimitador>" para "<nome_do_pseudônimo>". | O delimitador de colunas tinha mais de um caractere. Redefina a opção com um caractere simples. Em seguida, execute novamente o comando da instrução SQL. |
| SQL1882N | "KEY_COLUMN" da opção "Pseudônimo" não pode ser definido em "<nome_da_coluna>" para "<nome_do_pseudônimo>". | A coluna selecionada como coluna-chave não está definida para este pseudônimo. Corrija a opção KEY_COLUMN para que seja uma das colunas ordenadas para este pseudônimo e emita o comando SQL novamente. |

Tabela 22. Mensagens emitidas pelo wrapper para o arquivo estruturado de tabela (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL1882N | "VALIDATE_DATA_FILE" da opção "Pseudônimo" não pode ser definido em "<valor_da_opção>" para "<nome_do_pseudônimo>". | O valor da opção é inválido. Os valores válidos são "S" ou "N". Corrija a opção e registre o pseudônimo novamente. |
| SQL1883N | "<nome_da_opção>" é uma opção obrigatória do "<componente>" para "<nome_do_objeto>". | Estava faltando uma opção obrigatória da instrução SQL. Inclua-a e envie a instrução SQL novamente. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "21". | Você tentou acessar uma sessão de passagem. O wrapper do arquivo estruturado de tabela não suporta sessões de passagem. |

Related reference:

- "Mensagens para o Wrapper do Documentum" na página 194
- "Mensagens para o Wrapper do Excel" na página 211
- "Mensagens para o Wrapper do BLAST" na página 243
- "Mensagens para o Wrapper do XML" na página 268

Capítulo 13. Configurando o Acesso às Origens de Dados do Documentum

Este capítulo explica o que é o Documentum, como incluir origens de dados do Documentum em seu sistema federado e lista as mensagens de erro associadas ao wrapper do Documentum.

O que É Documentum?

Documentum é o software de gerenciamento de documentos que fornece gerenciamento de conteúdo de documentos e atributos, como registrar entrada, registrar saída, workflow e gerenciamento de versão. O produto Documentum é um sistema cliente-servidor de três camadas construído na parte superior de um banco de dados relacional.

Um Docbase é um repositório do Documentum que armazena conteúdo de documentos, atributos, relações, versões, traduções, formatos, workflow e segurança. Documentum Query Language (DQL), um dialeto SQL estendido, é utilizado para dados de consulta do Documentum. Um Docbase é o equivalente de uma instância do Oracle® ou de um banco de dados DB2® mais os arquivos de conteúdo de documentos. Os metadados são armazenados no sistema de gerenciamento do banco de dados relacional (relational database management system - RDBMS) subjacente e o conteúdo é armazenado como objetos binários extensos (binary large objects - BLOBs) no banco de dados ou como arquivos armazenados dentro do sistema de arquivos do sistema do servidor. Para obter maiores informações sobre o Documentum, consulte os manuais do Documentum.

O wrapper para Documentum permite incluir uma origem de dados do Documentum em um sistema federado do DB2. Ao incluir a origem de dados do Documentum em um sistema federado, você pode utilizar as instruções SQL para acessar objetos de consulta e tabelas registradas em um Documentum Docbase. Pode também integrar esses dados a outras origens de dados no sistema federado sem precisar remover os dados da origem de dados nativa. O wrapper do Documentum utiliza uma biblioteca do cliente para estabelecer interface com o servidor do Documentum. O wrapper do Documentum fornece acesso a duas versões do servidor Documentum: EDMS 98 (também referenciada como versão 3) e 4i. A Figura 2 na página 166 ilustra como o wrapper do Documentum funciona.

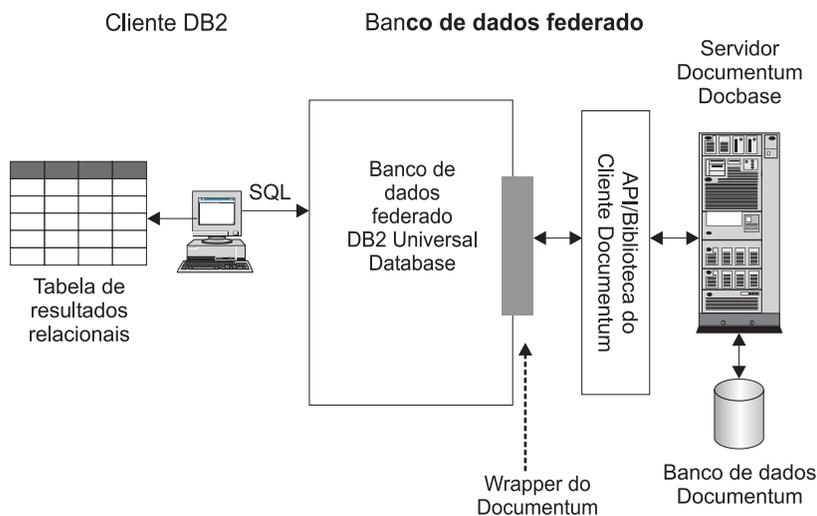


Figura 2. Como o wrapper do Documentum funciona

Depois que o wrapper do Documentum for registrado, você poderá mapear os objetos do Documentum Docbase e as tabelas registradas como tabelas relacionais. Isso é feito, mapeando os atributos do Docbase para nomes de colunas em uma tabela relacional do DB2.

Por exemplo, a Tabela 23 lista um subconjunto de atributos para o tipo de documento padrão do Documentum Docbase, `dm_document`, juntamente com os dados associados. Você determinou que esse subconjunto de atributos é importante e que gostaria de conectar esses atributos ao sistema de banco de dados federado. Esse subconjunto de dados recebeu o nome de `DrugAB_data`.

Tabela 23. `DrugAB_data`

| Título | Assunto | Autores | Palavras-chave |
|-------------------------------------|---------|-------------------------------|---------------------|
| O efeito da droga A nos coelhos | Droga A | Curran, L. | coelhos, droga A |
| Resultados da toxicidade da droga A | Droga A | Abelite, P., McMurtrey, K. | toxicidade, droga A |
| Interações da droga B | Droga B | DeNiro, R., Stone, S. | interações, droga B |
| Estrutura química da droga B | Droga B | Boyslim, F. | estrutura, droga B |

Depois de registrar o wrapper do Documentum, os dados poderão ser consultados, utilizando instruções SQL.

A consulta a seguir exibe os títulos e os autores cujo assunto é Droga A. A tabela de resultados é mostrada na Tabela 24.

```
SELECT title, authors
FROM drugAB_data
WHERE subject = 'Drug A'
```

Tabela 24. Resultados da consulta

| Título | Autores |
|-------------------------------------|----------------------------|
| O efeito da droga A nos coelhos | Curran, L. |
| Resultados da toxicidade da droga A | Abelite, P., McMurtrey, K. |

Related concepts:

- “O que São Arquivos com Estrutura de Tabela?” na página 151
- “O que É Excel?” na página 203
- “O que É BLAST?” na página 219
- “O que É XML?” na página 247

Related tasks:

- “Incluindo o Documentum em um Sistema Federado” na página 167

Incluindo o Documentum em um Sistema Federado

Procedimento:

Para incluir a origem de dados do Documentum para um servidor federado:

1. Disponibilize a biblioteca do cliente Documentum para o DB2.
2. Aponte para o arquivo dmcl.ini do cliente do Documentum
3. Registre o wrapper utilizando a instrução CREATE WRAPPER.Registre o wrapper utilizando a instrução CREATE WRAPPER.
4. Defina a variável de ambiente DB2_DJ_COMM para melhorar o desempenho da consulta.
5. Registre o servidor utilizando a instrução CREATE SERVER.Registre o servidor utilizando a instrução CREATE SERVER.
6. Forneça aos usuários acesso à origem de dados, utilizando a instrução CREATE USER MAPPING.
7. Registre pseudônimos utilizando a instrução CREATE NICKNAME.Registre pseudônimos utilizando a instrução CREATE NICKNAME.
8. Crie funções personalizadas, utilizando a instrução CREATE FUNCTION.

As instruções podem ser executadas a partir do Processador da Linha de Comandos do DB2. Depois de registradas, você poderá executar consultas na origem de dados.

Related tasks:

- “Disponibilizando a Biblioteca do Cliente Documentum para o DB2 (Apenas AIX e Solaris Operating Environment)” na página 168
- “Apontando para o Arquivo dmcl.ini do Cliente do Documentum” na página 169
- “Registrando o Wrapper do Documentum” na página 171
- “Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper Documentum” na página 171
- “Registrando o Servidor para Origens de Dados do Documentum” na página 172
- “Mapeando Usuários (Wrapper do Documentum)” na página 173
- “Registrando Pseudônimos para Origens de Dados do Documentum” na página 174
- “Registrando Funções Personalizadas para Origens de Dados do Documentum” na página 181

Disponibilizando a Biblioteca do Cliente Documentum para o DB2 (Apenas AIX e Solaris Operating Environment)

Disponibilizar a biblioteca do cliente Documentum para o DB2 (apenas AIX e Solaris Operating Environment) faz parte da maior tarefa de adicionar Documentum a um sistema federado.

A biblioteca do cliente deve estar disponível para o DB2 para que o wrapper funcione corretamente.

Pré-requisitos:

O wrapper Documentum utiliza a Versão 3.1.7a da biblioteca do cliente. Se você estiver utilizando o Documentum 4i, será necessário adquirir a versão antiga da biblioteca de cliente do Documentum (se ainda não estiver instalada).

Procedimento:

Para disponibilizar a biblioteca do cliente Documentum para o DB2, copie a biblioteca do cliente para o diretório apropriado. Consulte a Tabela 25 na página 169 para obter os nomes de bibliotecas do cliente e copiar para diretórios para cada sistema operacional suportado. Você também pode

efetuar um link simbólico para a biblioteca do cliente no diretório apropriado.

Tabela 25. Biblioteca do cliente e copiar para o diretório por sistema operacional

| Sistema Operacional | Biblioteca do Cliente | Copiar para o diretório |
|------------------------------|-----------------------|-------------------------|
| AIX | libdmcl.a | sql1lib/lib |
| Ambiente Operacional Solaris | libdmcl.so | sql1lib/lib |
| Windows | dmcl32.dll | x:\sql1lib\bin |

A próxima tarefa nesta seqüência de tarefas é apontar para o arquivo dmcl.ini do cliente Documentum.

Related tasks:

- “Apontando para o Arquivo dmcl.ini do Cliente do Documentum” na página 169

Apontando para o Arquivo dmcl.ini do Cliente do Documentum

Apontar para um arquivo dmcl.ini do cliente Documentum faz parte da maior tarefa de adicionar Documentum a um sistema federado. Os acessos ao Documentum Docbases são controlados pelo arquivo dmcl.ini do cliente Documentum. Uma instância DB2 deve ter suas variáveis de ambiente definidas para o arquivo dmcl.ini do cliente Documentum a fim de obter acesso a um Documentum Docbase.

Procedimento:

Para definir as variáveis de ambiente:

1. Edite o arquivo db2dj.ini e defina a variável de ambiente DOCUMENTUM ou DMCL_CONFIG.

O exemplo a seguir mostra como seria o aspecto dessas variáveis em sistemas operacionais Unix.

```
DOCUMENTUM=<path>
```

ou

```
DMCL_CONFIG=<path>/dmcl.ini
```

onde <path> é o diretório completo que contém o arquivo dmcl.ini que você deseja utilizar.

O caminho padrão para a localização do arquivo dmcl.ini do Documentum é /pkgs/documentum. Se ambas as linhas forem incluídas,

DMCL_CONFIG será utilizado. Em sistemas operacionais Windows, seria utilizada uma barra invertida em vez da barra para definir a localização do arquivo dmcl.ini.

Nos Ambientes Operacionais AIX e Solaris, o arquivo db2dj.ini está localizado no diretório \$HOME/sqllib/cfg.

No Windows, o arquivo db2dj.ini está em x:\sqllib\cfg onde x: representa a unidade onde o diretório sqllib está localizado.

Certifique-se de que o nome de um docbroker, ao qual todos os Docbases acessíveis para a instância DB2 sejam relatados, esteja especificado no arquivo dmcl.ini, conforme mostrado na Figura 3.

```
##### DOCUMENTUM CLIENT CONFIGURATION FILE #####
#
# Copyright Documentum 1994.
# Version 3.1 of the Documentum Server.
#
# A generated client init file for the Documentum Server.
#
# The only REQUIRED information in this file is the
# [DOCBROKER_PRIMARY] section and an entry for host.
# The host value should be the name of host on which
# your network wide DocBroker is running

[DOCBROKER_PRIMARY]
host = server16.comp2.big.com
```

Figura 3. Amostra do arquivo dmcl.ini com nome de docbroker especificado

2. Recicle a instância DB2, emitindo os seguintes comandos:

```
db2stop
db2start
```

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper Documentum.

Related tasks:

- “Registrando o Wrapper do Documentum” na página 171

Registrando o Wrapper do Documentum

O registro do wrapper Documentum faz parte da maior tarefa de adicionar Documentum a um sistema federado. Você deve registrar o wrapper para acessar uma origem de dados. Wrappers são mecanismos que os servidores federados utilizam para se comunicar com as origens de dados e recuperar dados delas. Os wrappers são instalados no sistema como arquivos de biblioteca.

Procedimento:

Para registrar o wrapper Documentum, envie a instrução CREATE WRAPPER.

Por exemplo, para criar um wrapper do Documentum no AIX denominado Dctm_Wrapper, a partir do arquivo de biblioteca padrão, libdb21sdctm.a, envie a seguinte instrução:

```
CREATE WRAPPER Dctm_Wrapper LIBRARY 'libdb21sdctm.a'  
    OPTIONS(DB2_FENCED 'N');
```

Para obter uma tabela de nomes de arquivos de biblioteca padrão para o wrapper Documentum por plataforma suportada, consulte a seção de tarefas relacionadas.

A próxima tarefa nesta seqüência de tarefas é definir a variável de ambiente DB2_DJ_COMM para o wrapper Documentum.

Related tasks:

- “Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper Documentum” na página 171
- “Depois de Instalar Wrappers Não Relacionais” no *DB2 Information Integrator Installation Guide*

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper Documentum

A definição da variável de perfil DB2_DJ_COMM do DB2 para o wrapper Documentum faz parte da maior tarefa de adicionar Documentum a um sistema federado. Para aprimorar o desempenho quando origens de dados Documentum são acessadas, defina a variável de perfil DB2_DJ_COMM do DB2. Esta variável determina se o servidor federado carregará o wrapper na inicialização.

Procedimento:

Para definir a variável de perfil DB2_DJ_COMM do DB2, envie o comando db2set com a biblioteca do wrapper que corresponde ao wrapper especificado na instrução CREATE WRAPPER associada.

Por exemplo:

```
db2set DB2_DJ_COMM='libdb2lsdctm.a'
```

Verifique se não existem espaços em nenhum lado do sinal de igual (=).

Existe overhead associado ao carregamento das bibliotecas de wrapper durante a inicialização do banco de dados. Para evitar este overhead, especifique apenas as bibliotecas que você pretende acessar.

A próxima tarefa nesta seqüência de tarefas é registrar o servidor para origens de dados Documentum.

Related tasks:

- “Registrando o Servidor para Origens de Dados do Documentum” na página 172

Registrando o Servidor para Origens de Dados do Documentum

O registro do servidor para origens de dados Documentum faz parte da maior tarefa de adicionar um Documentum a um sistema federado. Depois de registrar o wrapper, você deve registrar um servidor correspondente.

Restrições:

Todos os servidores em execução na mesma instância do DB2 devem compartilhar os mesmos parâmetros de configuração no arquivo dmcl.ini do Documentum.

Procedimento:

Para registrar o servidor do Documentum no sistema federado, utilize a instrução CREATE SERVER.

Por exemplo, suponha que haja um servidor chamado Dctm_Server1 para o wrapper Dctm_Wrapper criado na instrução CREATE WRAPPER associada. Suponha que o servidor contenha um Doabase executado no AIX e que utiliza o Oracle para armazenar dados. Para registrar o servidor, envie a seguinte instrução:

```
CREATE SERVER Dctm_Server1
  TYPE DCTM
  VERSION 3
  WRAPPER Dctm_Wrapper
  OPTIONS( NODE 'Dctm_Docbase',
    OS_TYPE 'AIX',
    RDBMS_TYPE 'ORACLE');
```

A próxima tarefa nesta seqüência de tarefas é mapear usuários.

Related tasks:

- “Mapeando Usuários (Wrapper do Documentum)” na página 173

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- “Argumentos e Opções da Instrução CREATE SERVER - Wrapper Documentum” na página 381

Mapeando Usuários (Wrapper do Documentum)

O mapeamento de usuários (wrapper Documentum) faz parte da maior tarefa de adicionar Documentum a um sistema federado. Você deve mapear usuários para os servidores definidos anteriormente para que eles tenham acesso à origem de dados.

Procedimento:

Para mapear usuários para seus servidores federados, utilize a instrução CRATE USER MAPPING.

Por exemplo, a instrução CREATE USER MAPPING a seguir mapeia o usuário Chuck para o usuário Charles no servidor Dctm_Server1.

```
CREATE USER MAPPING FOR Chuck SERVER Dctm_Server1
  OPTIONS(REMOTE_AUTHID 'Charles', REMOTE_PASSWORD 'Charles_pw');
```

Você também pode definir seu próprio mapeamento de usuário. No exemplo a seguir, USER é uma palavra-chave que significa o usuário atual e não um usuário chamado USER.

```
CREATE USER MAPPING FOR USER SERVER Dctm_Server1
  OPTIONS(REMOTE_AUTHID 'Lisa', REMOTE_PASSWORD 'Lisa_pw')
```

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para origens de dados Documentum.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados do Documentum” na página 174

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*
- “Opções da Instrução CREATE USER MAPPING - Wrapper Documentum” na página 384

Registrando Pseudônimos para Origens de Dados do Documentum

O registro de pseudônimos para origens de dados Documentum faz parte da maior tarefa de adicionar um Documentum a um sistema federado. Depois de registrar um servidor e de mapear suas informações de usuário para o servidor, é necessário registrar pseudônimos correspondentes. Pseudônimos são utilizados quando você faz referência a uma origem de dados do Documentum em uma consulta.

Restrições:

- Sessões de passagem não são suportadas.
- Para cada conexão a um banco de dados DB2 estabelecida por um aplicativo DB2, o wrapper do Documentum pode suportar um máximo de 10 sessões simultâneas do Documentum e cada sessão pode gerenciar simultaneamente até 10 consultas do Documentum. Um único aplicativo DB2 pode ter várias consultas em andamento simultaneamente; o tempo de operação de uma consulta começa quando ela é enviada ao DB2 e termina quando o cursor correspondente sobre o conjunto de resultados é fechado. Em um determinado momento, no conjunto inteiro de consultas em andamento nesse momento, não podem ser referidos mais de 10 pseudônimos de um servidor Documentum. Os pseudônimos mencionados em mais de uma consulta, ou referenciados várias vezes em uma única consulta, devem ser contados apenas uma vez em cada vez que aparecerem.

Procedimento:

Para registrar pseudônimos, utilize a instrução CREATE NICKNAME para criar um pseudônimo para cada Docbase de cada tipo de objeto ou tabela registrada de seu interesse.

Compreendendo Pseudo Colunas

A instrução CREATE NICKNAME também define 6 pseudo colunas. Estas colunas são utilizadas para acessar conteúdo de objetos e outras informações.

As pseudo colunas e suas definições estão listadas na Tabela 26 na página 175.

Tabela 26. Nomes e definições de pseudo colunas

| Nome da pseudo coluna | Definição |
|------------------------------|------------------|
| GET_FILE | VARCHAR (255) |
| GET_FILE_DEL | VARCHAR (255) |
| GET_RENDITION | VARCHAR (255) |
| GET_RENDITION_DEL | VARCHAR (255) |
| HITS | INTEGER |
| SCORE | DOUBLE |

Tabela 27 lista pseudo colunas para cláusulas SELECT.

Tabela 27. Pseudo colunas para cláusulas SELECT

| Nome da pseudo coluna | Descrição |
|------------------------------|--|
| GET_FILE | <p>Recupera o arquivo de conteúdo da linha atual além dos valores da coluna.</p> <p>A extensão do arquivo de conteúdo é seu formato de nome do Documentum. Se houver um arquivo com o mesmo nome, ele será sobreposto.</p> <p>GET_FILE tenta obter o formato base do objeto. Seu valor na linha é o nome completo do arquivo ou a cadeia "no_content."</p> <p>Por exemplo:</p> <pre>SELECT object_name, get_file FROM ...</pre> <p>O arquivo de conteúdo é colocado no diretório do servidor especificado pela opção CONTENT_DIR do Servidor. Ele também é colocado em um subdiretório denominado com o nome local do DB2. O subdiretório será criado, se ainda não existir.</p> <p>Sua extensão será a extensão DOS definida no Docbase para o tipo de formato do documento. Por exemplo, ".doc", para documentos do MS Word.</p> |
| GET_FILE_DEL | <p>Esta função é igual à GET_FILE exceto que GET_FILE_DEL primeiro exclui o arquivo recuperado para a linha anterior nesta consulta, se houver algum. Seu valor na linha é o nome completo do arquivo ou a cadeia "no_content."</p> |

Tabela 27. Pseudo colunas para cláusulas SELECT (continuação)

| Nome da pseudo coluna | Descrição |
|-----------------------|---|
| GET_RENDITION | <p>Recupera o arquivo de conteúdo dessa tradução, uma cópia do documento original em um formato diferente, para a linha atual além dos valores da coluna.</p> <p>A extensão do arquivo de conteúdo é seu formato de nome do Documentum. Se houver um arquivo com o mesmo nome, ele será sobreposto.</p> <p>Para especificar o formato de tradução, um predicado no formato DCTM.RENDITION_FORMAT(<format) = 1 deve ser especificado na cláusula WHERE.</p> <p>Por exemplo:</p> <pre>SELECT object_name, get_rendition FROM ...WHERE DCTM.RENDITION_FORMAT('pdf')=1</pre> <p>GET_RENDITION tenta obter a tradução nomeada do objeto. Seu valor na linha é o nome completo do arquivo ou a cadeia "no_content."</p> <p>O arquivo de conteúdo é colocado no diretório do servidor especificado pela opção CONTENT_DIR do Servidor. Ele também é colocado em um subdiretório denominado com o nome local do DB2. O subdiretório será criado, se ainda não existir.</p> <p>Sua extensão será a extensão DOS definida no Docbase para o tipo de formato do documento. Por exemplo, ".doc", para documentos do MS Word.</p> |
| GET_RENDITION_DEL | <p>Esta função é igual à GET_RENDITION exceto que GET_RENDITION_DEL primeiro exclui o arquivo recuperado para a linha anterior, se houver, nessa consulta. Seu valor na linha é o nome completo do arquivo ou a cadeia "no_content."</p> |

A Tabela 28 na página 177 lista pseudo colunas para cláusulas SELECT em consultas que contêm cláusulas de pesquisa.

Tabela 28. Pseudo colunas para cláusulas SELECT em consultas que contêm cláusulas de pesquisa

| Nome da pseudo coluna | Descrição |
|------------------------------|---|
| HITS | <p>Contém um número inteiro que representa o número de locais no documento nos quais os critérios de pesquisa foram atendidos.</p> <p>Por exemplo:</p> <pre>SELECT r_object_id, object_name, hits FROM std_doc WHERE DCTM.SEARCH_WORDS (''workflow'' OR ''flowchart'')=1</pre> <p>Para cada documento retornado, o número de ocorrências das palavras "workflow" e "flowchart" dentro do conteúdo do documento são somados e retornados como o valor HITS.</p> <p>A pseudo coluna HITS é apropriada quando os documentos possuem somente um arquivo de conteúdo. Esse é o caso típico. Esta pseudo coluna pode ser utilizada em uma qualificação de cláusula WHERE para uma instrução SELECT. No entanto, ela também deve ser especificada na cláusula SELECT.</p> |

Tabela 28. Pseudo colunas para cláusulas SELECT em consultas que contêm cláusulas de pesquisa (continuação)

| Nome da pseudo coluna | Descrição |
|-----------------------|---|
| SCORE | <p>Contém uma posição de relevância de um documento.</p> <p>Utilize esta pseudo coluna em conjunto com o operador de conceitos ACCRUE do Documentum. Ambos retornam um número que indica quantas das palavras especificadas foram encontradas em cada documento retornado.</p> <p>Por exemplo:</p> <pre>SELECT object_name, score FROM std_doc WHERE DCTM.SEARCH_TOPIC('<ACCRUE>("document","management","workflow")')=1 AND SCORE >=75</pre> <p>A instrução retorna todos os documentos que possuem duas ou três das palavras especificadas em seu conteúdo. Se um documento possuir apenas uma das palavras, será atribuído a ele um score de 50; portanto os critérios da cláusula WHERE falharão e não serão retornados. Se duas das três palavras forem encontradas, será atribuído a um documento um score de 75. Se as três palavras forem encontradas, o score do documento será 88.</p> <p>A pseudo coluna SCORE é utilizada para documentos que possuem um arquivo de conteúdo. Esse é o caso típico.</p> <p>SCORE pode estar em uma cláusula SELECT apenas se WHERE contiver uma função SEARCH_WORDS() ou SEARCH_TOPIC(). Em uma cláusula WHERE, ele é utilizado em conjunto com o operador de conceitos ACCRUE.</p> <p>Para obter informações sobre o operador de conceitos ACCRUE, consulte a documentação do Documentum.</p> |

A próxima tarefa nesta seqüência de tarefas é registrar funções personalizadas para origens de dados Documentum.

Related concepts:

- “O que É o Utilitário CreateNicknameFile para o Wrapper do Documentum?” na página 189

Related tasks:

- “Registrando Funções Personalizadas para Origens de Dados do Documentum” na página 181

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper Documentum” na página 360
- “Instrução CREATE NICKNAME - Exemplo para o Wrapper Documentum” na página 179

Instrução CREATE NICKNAME - Exemplo para o Wrapper Documentum

A instrução CREATE NICKNAME a seguir define o std_doc do pseudônimo. Std_doc está associado a um Documentum Docbase com um tipo de objeto dm_document. A Tabela 29 mapeia os atributos e os tipos de dados do Documentum para nomes de colunas relacionais e tipos de dados do DB2 que são utilizados para construir a instrução CREATE NICKNAME.

Tabela 29. Mapeamento de atributos do Documentum nas colunas do DB2 para o pseudônimo std_doc

| Nome do atributo do Documentum | Tipo de dados do Documentum | Nome da coluna do DB2 | Tipo de dados do DB2 | Repetições? | Pode ser anulada? |
|--------------------------------|-----------------------------|-----------------------|----------------------|-------------|-------------------|
| object_name | string(255) | object_name | varchar | Não | Não |
| r_object_id | ID | object_id | char(16) | Não | Não |
| r_object_type | string(32) | object_type | varchar | Não | Não |
| title | string(255) | title | varchar | Não | Não |
| subject | string(128) | subject | varchar | Não | Não |
| authors | string(32) | author | varchar | Sim | Sim |
| keywords | string(32) | keyword | varchar | Sim | Sim |
| r_creation_date | time | creation_date | timestamp | Não | Sim |
| r_modify_date | time | modified_date | timestamp | Não | Sim |
| a_status | string(16) | status | varchar | Não | Não |
| a_content_type | string(32) | content_type | varchar | Não | Não |
| r_content_size | double | content_size | integer | Não | Não |
| owner_name | string(32) | owner_name | varchar | Não | Sim |

A Tabela 30 na página 180 descreve cada atributo do Documentum utilizado no pseudônimo.

Tabela 30. Descrição de atributos do Documentum para o pseudônimo std_doc

| Nome do atributo do Documentum | Descrição |
|--------------------------------|--|
| object_name | O nome do objeto definido pelo usuário. |
| r_object_id | O único identificador de objetos para este objeto, definido durante a criação. |
| r_object_type | O tipo do objeto, definido quando o objeto é criado. |
| title | O título do objeto definido pelo usuário. |
| subject | O assunto do objeto definido pelo usuário. |
| authors | A lista de autores definida pelo usuário para o objeto. |
| keywords | A lista de palavras-chave definida pelo usuário para o objeto. |
| r_creation_date | A data e hora em que o objeto foi criado. |
| r_modify_date | A data e hora em que o objeto foi modificado pela última vez. |
| a_status | Definido pelo servidor quando uma tarefa do roteador é encaminhada. O valor é retirado dos valores atribuídos ao attached_task_status existentes no objeto roteador. |
| a_content_type | O formato de arquivo do conteúdo do objeto. |
| r_content_size | O número de bytes existentes no conteúdo. Para documentos com várias páginas, este atributo registra o tamanho do primeiro conteúdo associado ao documento. |
| owner_name | O nome do proprietário do objeto (o usuário que criou o objeto). |

A Tabela 29 na página 179 pode ser convertida para a instrução CREATE NICKNAME a seguir.

```
CREATE NICKNAME std_doc (
  object_name varchar(255) not null,
  object_id char(16) not null OPTIONS(REMOTE_NAME 'r_object_id'),
  object_type varchar(32) not null OPTIONS(REMOTE_NAME 'r_object_type'),
  title varchar(255) not null,
  subject varchar(128) not null,
  author varchar(32) OPTIONS(REMOTE_NAME 'authors', IS_REPEATING 'Y'),
  keyword varchar(32) OPTIONS(REMOTE_NAME 'keywords', IS_REPEATING 'Y'),
  creation_date timestamp OPTIONS(REMOTE_NAME 'r_creation_date'),
  modified_date timestamp OPTIONS(REMOTE_NAME 'r_modify_date'),
  status varchar(16) not null OPTIONS(REMOTE_NAME 'a_status'),
  content_type varchar(32) not null OPTIONS(REMOTE_NAME 'a_content_type'),
  content_size integer not null OPTIONS(REMOTE_NAME 'r_content_size'),
  owner_name varchar(32))
FOR SERVER Dctm_Server2 OPTIONS (REMOTE_OBJECT 'dm_document', IS_REG_TABLE 'N')
```

Depois de enviar a instrução `CREATE NICKNAME`, você pode utilizar o pseudônimo `std_doc` para consultar o sistema federado. Pode também unir o pseudônimo `std_doc` a outros pseudônimos e tabelas no sistema federado.

No catálogo, o número de colunas para este pseudônimo é 6 ou mais do que está sendo especificado na instrução `CREATE NICKNAME` devido às pseudo colunas.

É possível usar o utilitário `CreateNicknameFile` para mapear automaticamente os tipos de Documentum para tipos de DB2 e para criar uma instrução `CREATE NICKNAME` inicial.

Related concepts:

- “O que É o Utilitário `CreateNicknameFile` para o Wrapper do Documentum?” na página 189

Related tasks:

- “Registrando Pseudônimos para Origens de Dados do Documentum” na página 174

Related reference:

- “Sintaxe da Instrução `CREATE NICKNAME` - Wrapper Documentum” na página 360

Registrando Funções Personalizadas para Origens de Dados do Documentum

O registro de funções personalizadas para origens de dados Documentum faz parte da maior tarefa de adicionar Documentum a um sistema federado. Você deve utilizar a instrução `CREATE FUNCTION` para registrar várias funções personalizadas. Pode utilizar essas funções para acessar alguns dos recursos do Documentum, como o conteúdo do documento de recuperação e pesquisa do texto completo dentro das consultas.

As funções personalizadas para predicados estão relacionadas na Tabela 31 na página 183.

As referências à função `TOPIC` destinam-se à função do Documentum fornecida por terceiros, como parte de seu sistema de indexação de texto completo, da Verity, Inc

Restrições:

Uma vez que o DB2 não suporta o tipo Booleano, a maioria das funções personalizadas (exceto USER) utilizadas na cláusula WHERE devem fazer uma verificação "=1", pois essas funções são definidas para retornar um número inteiro.

Por exemplo,

```
"... WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1"
```

Procedimento:

Para registrar funções personalizadas, utilize a instrução CREATE FUNCTION.

Todas as funções personalizadas devem ser registradas com o nome de esquema DCTM. O nome completamente qualificado de cada função é DCTM.<function_name>.

O exemplo a seguir registra a função personalizada ANY_EQ.

```
CREATE FUNCTION DCTM.ANY_EQ (CHAR(), CHAR()) RETURNS INTEGER  
AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION
```

Cada função personalizada deve ser registrada apenas uma vez para cada banco de dados DB2 com o wrapper do Documentum instalado.

Para ajudá-lo a registrar funções personalizadas, o arquivo de amostra, create_function_mappings.ddl, é fornecido no diretório sqllib/samples/lifesci/dctm. Esse arquivo contém definições para cada função personalizada. Você pode executar esse arquivo ddl para registrar as funções personalizadas de cada banco de dados DB2 com o wrapper do Documentum instalado.

Regras de Argumento da Cadeia de Função Personalizada

Todos os argumentos transmitidos como cadeias devem estar de acordo com as seguintes regras:

- Cada cadeia é colocada entre aspas simples.
- Aspas simples dentro de cadeias são expressas por duas aspas simples.

Utilizando Funções Personalizadas em Consultas

Os exemplos a seguir ilustram a utilização das funções personalizadas em consultas.

Para exibir o nome e o autor do objeto a partir do pseudônimo std_doc para documentos que possuem um ou mais autores chamados 'Dave Winters':

```
SELECT object_name,authors FROM std_doc  
WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1
```

Para exibir o nome e o autor do objeto a partir do pseudônimo std_doc para documentos que possuem um ou mais autores chamados 'Dave Winters' ou 'Jon Doe':

```
SELECT object_name,authors FROM std_doc
WHERE DCTM.ANY_IN(authors,'Dave Winters','Jon Doe')=1
```

Para exibir o nome e o r_object_id do objeto e para recuperar o arquivo de conteúdo, a partir do pseudônimo std_doc para documentos que contêm cadeias como 'Dave Win%' na coluna de autores:

```
SELECT object_name, r_object_id, get_file FROM std_doc
WHERE DCTM.ANY_LIKE(authors,'Dave Win%')=1
```

Tabela de Função Personalizada

A Tabela 31 lista as funções personalizadas para predicados.

Tabela 31. Funções personalizadas para predicados

| Nome da função | Descrição |
|--------------------|--|
| ANY_EQ(arg1, arg2) | <p>Testa um atributo de repetição para qualquer valor igual ao valor especificado. Obtém dois argumentos obrigatórios:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2 Especifica o valor a ser comparado.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1</pre> |
| ANY_NE(arg1, arg2) | <p>Testa um atributo de repetição para qualquer valor diferente do valor especificado. Obtém dois argumentos obrigatórios:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2 Especifica o valor a ser comparado.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_NE(authors,'Dave Winters')=1</pre> |
| ANY_LT(arg1, arg2) | <p>Testa um atributo de repetição para qualquer valor menor que o valor especificado. Obtém dois argumentos obrigatórios:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2 Especifica o valor a ser comparado.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_LT(num_approvers,4)=1</pre> |

Tabela 31. Funções personalizadas para predicados (continuação)

| Nome da função | Descrição |
|--------------------|--|
| ANY_GT(arg1, arg2) | <p>Testa um atributo de repetição para qualquer valor maior que o valor especificado. Obtém dois argumentos obrigatórios:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2 Especifica o valor a ser comparado.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_GT(num_approvers,3)=1</pre> |
| ANY_LE(arg1, arg2) | <p>Testa um atributo de repetição para qualquer valor menor ou igual ao valor especificado. Obtém dois argumentos obrigatórios:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2 Especifica o valor a ser comparado.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_LE(num_approvers,2)=1</pre> |
| ANY_GE(arg1, arg2) | <p>Testa um atributo de repetição para qualquer valor maior ou igual ao valor especificado. Obtém dois argumentos obrigatórios:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2 Especifica o valor a ser comparado.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_GE(num_approvers,1)=1</pre> |

Tabela 31. Funções personalizadas para predicados (continuação)

| Nome da função | Descrição |
|----------------------------|--|
| ANY_IN(arg1, arg2 – arg11) | <p>Testa um atributo de repetição para qualquer um dos dez valores existentes em uma lista de valores especificados. Obtém de 3 a 11 argumentos do mesmo tipo de dados:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2–arg11 Especifica uma lista de valores separados por vírgula a serem comparados.</p> <p>Por exemplo: ... WHERE DCTM.ANY_IN(authors,'Crick','Watson')=1</p> <p>O número máximo de valores em uma função personalizada ANY_IN para atributos de repetição é 10 para uma única instrução. Várias instruções podem ser OR'd.</p> |
| ANY_LIKE(arg1, arg2) | <p>Testa um atributo de repetição para qualquer valor igual ao valor especificado. Obtém dois argumentos obrigatórios:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2 Especifica o padrão que está sendo comparado com subcadeias entre aspas simples.</p> <p>Por exemplo: ... WHERE DCTM.ANY_LIKE(authors,'Dave Win%')=1 OR DCTM.ANY_LIKE(keywords,'%_%')=1</p> <p>A cláusula de escape não é suportada em predicados ANY_LIKE().</p> |

Tabela 31. Funções personalizadas para predicados (continuação)

| Nome da função | Descrição |
|------------------------------|--|
| ANY_NOT_LIKE(arg1, arg2) | <p>Testa um atributo de repetição para qualquer valor diferente do valor especificado. Obtém dois argumentos obrigatórios:</p> <p>arg1 Especifica o nome de uma coluna que representa um atributo de repetição.</p> <p>arg2 Especifica o padrão que está sendo comparado com subcadeias entre aspas simples.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_NOT_LIKE(authors, 'Dave Win%')=1 OR DCTM.ANY_NOT_LIKE(keywords, '%_%')=1</pre> <p>A cláusula de escape não é suportada em predicados ANY_NOT_LIKE().</p> |
| ANY_NULL(arg) | <p>Testa um atributo de repetição para IS NULL. Obtém um argumento obrigatório que é o nome do atributo de repetição ou o atributo DATE ou TIMESTAMP de valor único.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_NULL(authors)=1</pre> |
| ANY_NOT_NULL(arg) | <p>Testa um atributo de repetição para NOT NULL. Obtém um argumento obrigatório que é o nome do atributo de repetição.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.ANY_NOT_NULL(authors)=1</pre> |
| ANY_SAME_INDEX(arg1 – arg10) | <p>Testa os atributos de repetição para valores no mesmo índice de cada atributo. Obtém de duas a dez das outras funções ANY_xx().</p> <p>O exemplo a seguir verifica se um documento possui pelo menos um autor chamado Ken que não está associado ao UCD.</p> <pre>... WHERE DCTM.ANY_SAME_INDEX(ANY_EQ(author_name, 'Ken'), DCTM.ANY_NE(author_affiliation, 'UCD'))=1</pre> <p>O número máximo de testes para valores no mesmo índice de atributos de repetição é 10. Os testes devem ser testes AND que são avaliados da esquerda para a direita.</p> |

Tabela 31. Funções personalizadas para predicados (continuação)

| Nome da função | Descrição |
|-------------------------------------|---|
| CABINET(arg) e CABINET_TREE(arg) | <p>Obtém um argumento obrigatório que é o nome completamente qualificado de um gabinete do Docbase.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.CABINET('/Tools')=1 ... WHERE DCTM.CABINET_TREE('/MyDocs')=1</pre> <p>Utilize várias instâncias de CABINET e CABINET_TREE para especificar vários gabinetes.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.CABINET('/Tools')=1 OR DCTM.CABINET_TREE('/Parts')=1</pre> |
| FOLDER(arg) e FOLDER_TREE(arg) | <p>Obtém um argumento obrigatório que é o nome completamente qualificado de uma pasta ou gabinete do Docbase.</p> <p>Por exemplo:</p> <pre>... DCTM.FOLDER('/Tools/Drills')=1 ... DCTM.FOLDER_TREE('/MyDocs/WhitePapers')=1</pre> <p>Utilize várias instâncias de FOLDER e FOLDER_TREE para especificar várias pastas.</p> <p>Por exemplo:</p> <pre>... DCTM.FOLDER('/Tools/Drills')=1 OR DCTM.FOLDER_TREE('/Animals/Horses')=1</pre> |
| RENDITION_FORMAT (format) | <p>Trabalha com as pseudo colunas GET_RENDITION e GET_RENDITION_DEL para estabelecer o formato da tradução a ser recuperado. Aceita um único argumento de cadeia de caracteres para especificar o formato.</p> <p>O exemplo a seguir recupera um documento no formato PDF:</p> <pre>SELECT get_rendition FROM WHERE DCTM.RENDITION_FORMAT('pdf')=1</pre> |

Tabela 31. Funções personalizadas para predicados (continuação)

| Nome da função | Descrição |
|-------------------|---|
| USER(1) | <p>Compara um valor com a ID de autor Documentum do usuário atual. Devido à limitação do DB2, a função personalizada USER é definida com um argumento inteiro que não é utilizado.</p> <p>Por exemplo:</p> <pre>... WHERE approver = DCTM.USER(1)</pre> <p>Para que a ID de autor Documentum corresponda à ID de autor do DB2, utilize a instrução CREATE USER MAPPING.</p> |
| SEARCH_WORDS(arg) | <p>Obtém um argumento de cadeia obrigatório que é uma lista de palavras individuais colocadas entre aspas simples, separadas por AND, OR ou NOT e que utiliza parênteses para controlar precedência. As palavras não podem conter espaços em branco e devem ser colocadas entre aspas simples.</p> <p>Por exemplo:</p> <pre>... DCTM.SEARCH_WORDS(''yeast'' AND (''bread'' OR ''cake'') AND NOT ''wedding'')=1</pre> |
| SEARCH_TOPIC(arg) | <p>Obtém um argumento de cadeia obrigatório que é a instrução de consulta TOPIC da Verity a ser transmitida literalmente ao Documentum e ao Verity.</p> <p>Por exemplo:</p> <pre>... WHERE DCTM.SEARCH_TOPIC('"quick"')=1</pre> |

Não há mais tarefas nesta seqüência de tarefas.

Related reference:

- “CREATE FUNCTION (Sourced or Template) statement” no *SQL Reference, Volume 2*

Origem de Dados de Documentum – Consultas de Exemplo

Depois de registrar o wrapper, você poderá executar consultas SQL na origem de dados Documentum. Esta seção fornece vários exemplos de consultas.

Para executar consultas, você utiliza o pseudônimo e as colunas de pseudônimo definidas em suas instruções SQL da mesma maneira que utilizaria um nome de tabela e colunas de tabela regulares.

O servidor Documentum e o DB2 processam o predicado LIKE de forma diferente. Quando é efetuado pushdown de um predicado LIKE para o servidor Documentum, as semânticas do Documentum se aplicam. No exemplo a seguir, quando a coluna c1 contiver uma cadeia de comprimento zero, o predicado será true para Documentum e false para o DB2.

```
c1 LIKE '%'
```

A consulta a seguir exibe todos os documentos do Docbase para documentos chamados 'Test Document':

```
SELECT object_name
FROM std_doc
WHERE object_name='Test Document';
```

A consulta a seguir utiliza a função personalizada ANY_EQ para exibir todos os documentos em que um dos autores é 'Joe Doe'.

```
SELECT object_name
FROM std_doc
WHERE DCTM.ANY_EQ(author, 'Joe Doe')=1
```

A consulta a seguir utiliza a função FOLDER_TREE e a função SEARCH_WORDS para localizar todos os documentos no gabinete Approved que contêm o texto "protein".

```
SELECT object_name
FROM std_doc
WHERE DCTM.FOLDER_TREE('/Approved')=1
      AND DCTM.SEARCH_WORDS('protein')=1
```

A consulta a seguir utiliza a pseudo coluna GET_FILE e as funções personalizadas FOLDER_TREE e ANY_IN para recuperar o nome dos arquivos no servidor DB2, nos quais foi colocado o conteúdo para todos os documentos existentes no gabinete Approved que possuem qualquer um dos autores relacionados.

```
SELECT object_name, object_id, get_file
FROM std_doc
WHERE DCTM.FOLDER_TREE('/Approved')=1
      AND DCTM.ANY_IN(author, 'Mary Black', 'Joe Carson', 'Peter Miller')=1
```

Related reference:

- "Origem de Dados Excel – Consultas de Exemplo" na página 207

O que É o Utilitário CreateNicknameFile para o Wrapper do Documentum?

Você pode usar um utilitário do Docbasic chamado CreateNicknameFile, disponível para download grátis, para criar um arquivo ASCII que contém uma definição completa de qualquer objeto do Docbase ou tabela registrada. Pode editar o arquivo de saída para:

- Definir nomes locais personalizados para colunas e atributos. Os nomes locais e remotos são inicialmente aqueles conhecidos no Docbase.
- Excluir as colunas e os atributos indesejados. O único tipo de documento do Documentum pré-definido (dm_document) possui 59 atributos em EDMS98 e 76 atributos em 4i. A maioria deles contém metadados para gerenciamento de documentos de nível inferior e desenvolvimento de aplicativos. A exclusão dos atributos que não interessam mais pode tornar as instruções SELECT * SQL mais úteis, sem resultar em um impacto no desempenho.
- Adicionar um valor para a opção FOLDERS para restringir pesquisas neste pseudônimo para pastas específicas do Documentum.
- Alterar os mapeamentos de DATE para TIMESTAMP, se o desejar. O utilitário gera um mapeamento a partir do DQL DATE para DB2® DATE porque parece ser mais útil.
- Alterar os mapeamentos CHAR para VARCHAR ou vice-versa dependendo do aplicativo interno.

Você deve instalar o utilitário em um Docbase e executá-lo a partir de uma interface gráfica com o usuário Windows® do Documentum. Os arquivos que o utilitário gera são específicos do Docbase em que ele está instalado.

Related tasks:

- “Instalando o Utilitário CreateNicknameFile (Wrapper do Documentum)” na página 190
- “Configurando o Utilitário CreateNicknameFile (Wrapper do Documentum)” na página 191
- “Mapeando o Tipo de Objeto DM_ID em Tabelas Registradas do Documentum” na página 192

Instalando o Utilitário CreateNicknameFile (Wrapper do Documentum)

O utilitário CreateNicknameFile pode ajudá-lo a gravar instruções CREATE NICKNAME para suas origens de dados do Documentum.

Procedimento:

Para instalar o utilitário:

1. Faça download do utilitário CreateNicknameFile a partir da seção de download do Web site do produto DB2 Information Integrator.
2. Utilize a interface gráfica com o usuário EDMS98 Workspace ou 4i Desktop Client para importar o utilitário, chamado

CreateNicknameFile.txt. Você pode importar o utilitário como um tipo de procedimento para qualquer gabinete ou pasta do Docbase e fornecer-lhe qualquer nome que desejar.

3. Marque a caixa **Can be run by user** no diálogo de propriedades para o objeto CreateNicknameFile.txt importado recentemente.

Related concepts:

- “O que É o Utilitário CreateNicknameFile para o Wrapper do Documentum?” na página 189

Related tasks:

- “Configurando o Utilitário CreateNicknameFile (Wrapper do Documentum)” na página 191
- “Mapeando o Tipo de Objeto DM_ID em Tabelas Registradas do Documentum” na página 192

Configurando o Utilitário CreateNicknameFile (Wrapper do Documentum)

O utilitário CreateNicknameFile pode ajudá-lo a gravar instruções CREATE NICKNAME para suas origens de dados do Documentum.

Pré-requisitos:

Você deve instalar o utilitário CreateNicknameFile para que ele possa ser configurado.

Procedimento:

Para configurar o utilitário depois de instalá-lo:

1. Clique duas vezes no ícone do utilitário para executá-lo.
2. Digite o nome do Documento/tipo de objeto do Documentum. O padrão é dm_document.

Especifique dm_registered como o nome se precisar criar um arquivo de pseudônimo para uma tabela registrada. Se especificar dm_registered, será solicitado também o nome da tabela completamente qualificado no formato <owner>.<table_name>. dm_dbo pode ser utilizado como o nome do proprietário se a tabela pertencer ao proprietário do Docbase (o caso típico).

O utilitário assume uma convenção de nomenclatura para os nomes de pseudônimos em tabelas registradas. A convenção é para prefixar o nome da tabela com "rt_" para indicar "tabela registrada" ("registered table"). Você pode alterar o pseudônimo proposto pelo utilitário se não deseja utilizar essa convenção.

3. Digite o nome do servidor associado ao pseudônimo que você está criando.
4. Digite o nome do pseudônimo.
Os nomes do pseudônimo devem ser evidentes e devem ser exclusivos na instância do DB2. O utilitário assume uma convenção de nomenclatura de <server_name>.<object_type>, porque talvez o mesmo <object_type> precise ser definido para vários servidores. Você pode alterar o pseudônimo proposto pelo utilitário se não deseja seguir essa convenção.
5. Digite o nome do arquivo de saída.
O padrão é C:\Temp\nickname.txt. O diretório que receberá o arquivo de saída deve existir e ser gravável por você.

Depois que você responder aos avisos, o arquivo de pseudônimo será criado e aberto em um editor de texto.

Related concepts:

- “O que É o Utilitário CreateNicknameFile para o Wrapper do Documentum?” na página 189

Related tasks:

- “Registrando Pseudônimos para Origens de Dados do Documentum” na página 174
- “Instalando o Utilitário CreateNicknameFile (Wrapper do Documentum)” na página 190

Mapeando o Tipo de Objeto DM_ID em Tabelas Registradas do Documentum

As definições da coluna criadas pelo utilitário CreateNicknameFile são compatíveis com os requisitos do wrapper do Documentum, incluindo o mapeamento correto de cada tipo de dados para o tipo de dados DB2 correspondente. A única exceção é que o Documentum não suporta o tipo de dados DM_ID em tabelas registradas. O utilitário assume a utilização de uma coluna em uma tabela registrada para conter uma ID do objeto se for definido com uma cadeia, possuir 16 caracteres de comprimento e possuir um nome terminando com "_id". No caso do tipo de dados DM_ID, o utilitário mapeia a coluna para o tipo de dados DB2 CHAR(16). Em todos os outros casos, todas as colunas string/varchar são mapeadas para o tipo de dados DB2 VARCHAR.

Procedimento:

Para assegurar mapeamento do tipo de dados correto:

1. Examine as definições do tipo de dados da coluna no arquivo de saída criado pelo utilitário CreateNicknameFile.
2. Se o utilitário mapear um tipo de dados de uma coluna do Documentum para um tipo de dados DB2 incorreto, mude o tipo de dados DB2 antes de utilizar o arquivo para registrar o pseudônimo para o DB2.

Related concepts:

- “O que É o Utilitário CreateNicknameFile para o Wrapper do Documentum?” na página 189

Related tasks:

- “Instalando o Utilitário CreateNicknameFile (Wrapper do Documentum)” na página 190
- “Configurando o Utilitário CreateNicknameFile (Wrapper do Documentum)” na página 191

Definição Dupla dos Atributos de Repetição (Wrapper do Documentum)

Para maximizar os recursos da consulta do wrapper, cada atributo deve ser definido como seu verdadeiro tipo de dados DB2 equivalente. Isto é, os números inteiros do Documentum devem ser definidos como inteiros do DB2 e assim por diante. No entanto, essas definições evitam o retorno de diversos valores para atributos de repetição não-VARCHAR. Para tais colunas, apenas o último valor é retornado.

Esta restrição existe porque, sempre que possível, o wrapper retorna apenas uma linha de resultados por objeto Docbase. Essa restrição só é uma emissão quando os atributos de repetição são selecionados. No entanto, você pode definir uma segunda coluna para o mesmo atributo de repetição remoto, porém com um tipo de dados de VARCHAR.

Esse nome de coluna será usado na lista SELECT para retornar todos os valores como uma lista de todos os seus valores separada por delimitadores. (Cada opção DELIMITER da coluna especifica o delimitador a ser utilizado.)

Você deve padronizar os nomes dos locais das colunas multivalores. Você pode padronizar os nomes dos locais de cada coluna multivalor, incluindo um prefixo "m_" no nome do local da coluna que foi definida como seu verdadeiro tipo de dados.

Por exemplo, suponha que você tenha uma coluna de pseudônimos de um atributo de repetição do Documentum denominada approval_dates definida com o tipo de dados TIMESTAMP. Pode-se criar uma segunda coluna de pseudônimos denominada m_approval_dates e defini-la como um tipo de

dados VARCHAR. Em seguida, pode-se usar `m_approval_dates` em uma lista `SELECT` para retornar todas as datas de aprovação em uma lista separada por delimitadores.

Não é necessário usar definições duplas para atributos de repetição cujo verdadeiro tipo de dados é VARCHAR.

Controle de Acesso para o Wrapper do Documentum

As consultas estão sujeitas às permissões no Docbase. Apenas os documentos aos quais você tem, pelo menos, acesso de leitura são incluídos nos resultados da consulta.

Related reference:

- “Modelo de Controle de Acesso de Arquivo para o Wrapper do Arquivo Estruturado de Tabela” na página 159
- “Modelo de Controle de Acesso de Arquivo para o Wrapper do Excel” na página 210

Mensagens para o Wrapper do Documentum

Esta seção lista e descreve mensagens que podem ser encontradas enquanto se trabalha com o wrapper do Documentum.

Tabela 32. Mensagens emitidas pelo wrapper para o Documentum

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "sqlno_crule_save_plans [100]:rc (-2144272209) Foi dectada uma lista de plano vazia".) | A consulta SQL enviada para o DB2 não foi processada pelo wrapper. Corrija a sintaxe e envie novamente. |

Tabela 32. Mensagens emitidas pelo wrapper para o Documentum (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Falha no dmAPI exec: erro [DM_QUERY_E_BAD_QUAL]: "O qualificador de atributos, A0, para o atributo <nome-da-coluna>, não é um qualificador válido."".) | Foi digitado, incorretamente, um tipo ou uma tabela registrada do Documentum para a opção de pseudônimo REMOTE_OBJECT. Altere o pseudônimo para utilizar o tipo de objeto ou a tabela registrada corretos do Documentum. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Coluna nula especificada inválida".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "A especificação de pseudônimo está em branco".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "O objeto Output está vazio ou incompleto".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Número de colunas solicitadas não esperado".) | Erro de programação interna. Contate o Suporte de Software da IBM. |

Tabela 32. Mensagens emitidas pelo wrapper para o Documentum (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Nenhuma informação de coluna encontrada".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Tipo de coluna solicitado não suportado".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Definição de coluna incorreta".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Tipo inconsistente; solicitação DB2 != tipo de pseudônimo".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "O parâmetro Output não é NULL".) | Erro de programação interna. Contate o Suporte de Software da IBM. |

Tabela 32. Mensagens emitidas pelo wrapper para o Documentum (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|--|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "A variável de resultado da consulta não é NULL".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Comprimento do registro de tempo inválido".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Número de colunas inconsistente".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Impossível não acessar dados na conversão de valores".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Falha ao inicializar o cliente DMCL".) | O cliente Documentum não pode ser inicializado. Entre em contato com o administrador do sistema. |

Tabela 32. Mensagens emitidas pelo wrapper para o Documentum (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Get_User retornou NULL".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Get_Local_User retornou NULL".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Falha ao iniciar a transação".) | O Documentum informou que o begintrans falhou. Entre em contato com o administrador do sistema. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "O parâmetro de entrada não era NULL".) | Erro de programação interna. Contate o Suporte de Software da IBM. |
| SQL901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Funções Dctm devem ser iguais a DCTM.function(...) =1".) | Você não utilizou =1 como o RHS do predicado para uma função Dctm. Corrija a sintaxe e execute a consulta novamente. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Número de colunas solicitadas inválido".) | Erro de programação interna. Contate o Suporte de Software da IBM. |

Tabela 32. Mensagens emitidas pelo wrapper para o Documentum (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|--|
| SQL1881N | "DELIMITER" não é uma opção "COLUMN" válida para "<nome-da-coluna>" | A opção DELIMITER foi especificada para a coluna <nome-da-coluna>, mas a opção IS_REPEATING não foi especificada. |
| SQL1882N | A opção "RDBMS_TYPE" para "SERVER" não pode ser definida para "<valor-da-opção>" para "<nome-do-servidor>". | O valor especificado para a opção de servidor RDBMS_TYPE é inválido. Ele deve ser um dos seguintes: DB2, INFORMIX, ORACLE, SQLSERVER ou SYBASE. |
| SQL1882N | A opção "TRANSACTIONS" para "SERVER" não pode ser definida para "<valor-da-opção>" para "<nome-do-servidor>". | O valor especificado para a opção de servidor TRANSACTIONS é inválido. Ele deve ser um dos seguintes: NONE, QUERY, PASSTHRU ou ALL. |
| SQL1882N | A opção "IS_REG_TABLE" para "NICKNAME" não pode ser definida para "<valor-da-opção>" para "<pseudônimo>". | O valor especificado para a opção de pseudônimo IS_REG_TABLE é inválido. Ele deve ser um dos seguintes: 'Y' ou 'N'. |
| SQL1882N | A opção "ALL_VERSIONS" para "NICKNAME" não pode ser definida para "<valor-da-opção>" para "<pseudônimo>". | O valor especificado para a opção de pseudônimo ALL_VERSIONS é inválido. Ele deve ser um dos seguintes: 'Y' ou 'N'. |
| SQL1882N | A opção "OS_TYPE" para "SERVER" não pode ser definida para "<valor-da-opção>" para "<nome-do-servidor>" | O valor especificado para a opção de servidor OS_TYPE é inválido. Ele deve ser: AIX, HPUX, SOLARIS ou WINDOWS. |
| SQL1882N | A opção "FOLDERS" para "NICKNAME" não pode ser definida para "<valor-da-opção>" para "<pseudônimo>" | O valor especificado para a opção de pseudônimo FOLDERS é inválido. Ele não pode ser especificado para uma tabela onde IS_REG_TABLE seja 'Y'. |
| SQL1882N | A opção "VERSIONS" para "NICKNAME" não pode ser definida para "<valor-da-opção>" para "<pseudônimo>" | O valor especificado para a opção de pseudônimo VERSIONS é inválido. Ele deve ser um dos seguintes: 'Y' ou 'N'. Além disso, VERSIONS 'Y' não pode ser especificado para uma tabela onde IS_REG_TABLE seja 'Y'. |

Tabela 32. Mensagens emitidas pelo wrapper para o Documentum (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Nome de coluna inválido, IS_REG_TABLE ou IS_REPEATING especificado no pseudônimo" | Verifique a instrução de pseudônimo para obter a especificação correta das opções IS_REG_TABLE, IS_REPEATING, REMOTE_NAME e os nomes das colunas. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "db2dj.ini faltando na variável de ambiente DOCUMENTUM ou DMCL_CONFIG" | As variáveis de ambiente obrigatórias não estão definidas. Defina-as no arquivo db2dj.ini. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Falha ao abrir o arquivo de log para depuração" | O arquivo de log utilizado para detecção de problemas não está acessível. Entre em contato com o administrador do sistema. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Somente uma condição de pesquisa pode ser especificada" | Somente uma função de pesquisa personalizada pode ser especificada por consulta. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Falha ao criar diretório de conteúdo" | Verifique se o diretório de destino pode ser gravado pelo agente DB2. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Falha ao alterar as permissões no arquivo de conteúdo" | Verifique se o diretório do conteúdo de destino pode ser gravado pelo agente db2. |

Related reference:

- "Mensagens para o Wrapper do Arquivo Estruturado de Tabela" na página 160
- "Mensagens para o Wrapper do Excel" na página 211

- “Mensagens para o Wrapper do BLAST” na página 243
- “Mensagens para o Wrapper do XML” na página 268

Capítulo 14. Configurando o Acesso às Origens de Dados do Excel

Este capítulo explica o que é o Excel, como incluir origens de dados do Excel em seu sistema federado e lista as mensagens de erro associadas ao wrapper do Excel.

O que É Excel?

Uma planilha ou uma pasta do Excel é um arquivo criado com o aplicativo Microsoft® (MS) Excel e possui uma extensão de arquivo xls. O DB2® Information Integrator suporta planilhas do Excel 97 e Excel 2000. A Figura 4 ilustra como o wrapper do Excel conecta as planilhas ao sistema federado.

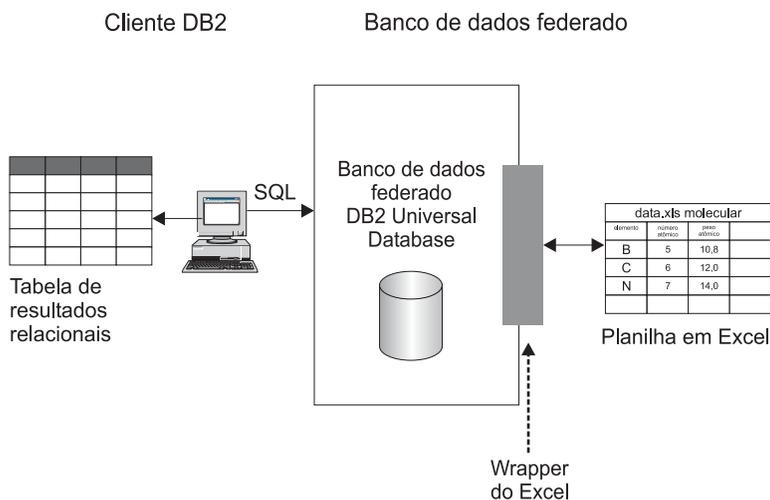


Figura 4. Como o wrapper do Excel funciona

O wrapper Excel utiliza a instrução CREATE NICKNAME para mapear as colunas na planilha Excel para as colunas no sistema federado DB2. A Tabela 33 mostra dados da planilha de amostra que é armazenada em um arquivo chamado Compound_Master.xls.

Tabela 33. Planilha de amostra para Compound_Master.xls

| | A | V | C | D |
|---|------------|------|-----|---------|
| 1 | compound_A | 1.23 | 367 | testado |

Tabela 33. Planilha de amostra para Compound_Master.xls (continuação)

| | A | V | C | D |
|---|------------|-------------|------|---------|
| 2 | compound_G | | 210 | |
| 3 | compound_F | 0.000425536 | 174 | testado |
| 4 | compound_Y | 1.00256 | | testado |
| 5 | compound_Q | | 1024 | |
| 6 | compound_B | 33.5362 | | |
| 7 | compound_S | 0.96723 | 67 | testado |
| 8 | | | | |
| 9 | compound_O | 1.2 | | testado |

Essas informações normalmente não estão disponíveis através dos comandos SQL padrão. Quando o wrapper do Excel é instalado e registrado, você pode acessar essas informações como se fosse uma origem de dados relacional padrão. Por exemplo, se você deseja conhecer todos os dados compostos onde a contagem molecular é maior que 100, execute a seguinte consulta SQL:

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

Os resultados da consulta são mostrados na Tabela 34.

Tabela 34. Resultados da consulta

| COMPOUND_NAME | WEIGHT | MOL_COUNT | WAS_TESTED |
|---------------|-------------|-----------|------------|
| compound_A | 1.23 | 367 | testado |
| compound_G | | 210 | |
| compound_F | 0.000425536 | 174 | testado |
| compound_Q | | 1024 | |

Related concepts:

- “O que São Arquivos com Estrutura de Tabela?” na página 151
- “O que É Documentum?” na página 165
- “O que É BLAST?” na página 219
- “O que É XML?” na página 247

Related tasks:

- “Incluindo Excel a um Sistema Federado” na página 205

Incluindo Excel a um Sistema Federado

Procedimento:

Para incluir a origem de dados do Excel a um sistema federado:

1. Registre o wrapper utilizando a instrução CREATE WRAPPER. Registre o wrapper utilizando a instrução CREATE WRAPPER.
2. Registre o servidor utilizando a instrução CREATE SERVER. Registre o servidor utilizando a instrução CREATE SERVER.
3. Registre os pseudônimos utilizando a instrução CREATE NICKNAME para cada planilha do Excel que você deseja acessar.

Os comandos podem ser executados a partir do Processador da Linha de Comandos do DB2.

Related tasks:

- “Registrando o Wrapper do Excel” na página 205
- “Registrando o Servidor para uma Origem de Dados do Excel” na página 206
- “Registrando Pseudônimos para Origens de Dados do Excel” na página 207

Registrando o Wrapper do Excel

O registro do wrapper Excel faz parte da maior tarefa de adicionar Excel a um sistema federado. Você deve registrar o wrapper para acessar uma origem de dados. Wrappers são mecanismos que os servidores federados utilizam para se comunicar com as origens de dados e recuperar dados delas. Os wrappers são instalados no sistema como arquivos de biblioteca.

Restrições:

- Os wrappers do Excel estão disponíveis apenas para os sistemas operacionais Microsoft Windows que suportam DB2 Universal Database Enterprise Server Edition.
- O aplicativo MS Excel deve ser instalado no servidor no qual o DB2 Information Integrator está instalado antes da utilização de um wrapper Excel.
- Sessões de passagem não são permitidas.

Procedimento:

Para registrar o wrapper da origem de dados do Excel, envie uma instrução CREATE WRAPPER.

Para criar um wrapper do Excel para o Excel 97 denominado Excel_9x_Wrapper utilizando o arquivo de biblioteca db21sx1s.dll, envie a seguinte instrução:

```
CREATE WRAPPER Excel_9x_Wrapper LIBRARY 'db21sx1s.dll'  
  OPTIONS(DB2_FENCED 'N');
```

A próxima tarefa nesta seqüência de tarefas é registrar o servidor para uma origem de dados Excel.

Related tasks:

- “Registrando o Servidor para uma Origem de Dados do Excel” na página 206

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Registrando o Servidor para uma Origem de Dados do Excel

O registro do servidor para uma origem de dados Excel faz parte da maior tarefa de adicionar Excel a um sistema federado. Depois de registrar o wrapper, você deve registrar um servidor correspondente.

Procedimento:

Para registrar o servidor do Excel no sistema federado, utilize a instrução CREATE SERVER.

Por exemplo, para criar um servidor chamado biochem_lab, com um nome de nó biochem_node1 que registra o servidor para o wrapper Excel_2000_Wrapper criado utilizando a instrução CREATE WRAPPER, envie a seguinte instrução:

```
CREATE SERVER biochem_lab WRAPPER Excel_2000_Wrapper;
```

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para origens de dados Excel.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados do Excel” na página 207

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- “Argumentos da Instrução CREATE SERVER - Wrapper Excel” na página 383

Registrando Pseudônimos para Origens de Dados do Excel

O registro de pseudônimos para origens de dados Excel faz parte da maior tarefa de adicionar Excel a um sistema federado. Depois de registrar o servidor, você deve registrar um pseudônimo correspondente. Pseudônimos são utilizados quando você faz referência a uma origem de dados do Excel em uma consulta.

Restrições:

O intervalo de data suportado pelo wrapper do tipo de dados DATE é de 1 de janeiro de 1970 a 18 de janeiro de 2038.

Procedimento:

Para mapear a origem de dados do Excel nas tabelas relacionais, crie um pseudônimo utilizando a instrução CREATE NICKNAME.

A instrução contida no exemplo a seguir cria o pseudônimo Compounds do arquivo da planilha do Excel denominado CompoundMaster.xls. O arquivo contém três colunas de dados que estão sendo definidas para o sistema federado como Compound_ID, CompoundName e MolWeight.

```
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:E5');
```

Não há mais tarefas nesta seqüência de tarefas.

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper Excel” na página 364

Origem de Dados Excel – Consultas de Exemplo

Este tópico lista várias consultas de planilhas Excel de amostra utilizando o pseudônimo Compounds.

Para executar consultas, você utiliza o pseudônimo e as colunas de pseudônimo definidas em suas instruções SQL da mesma maneira que utilizaria um nome de tabela e colunas de tabela regulares.

A consulta a seguir exibe todos os compound_IDs em que o peso molecular é maior que 2000:

```
SELECT compound_ID
FROM Compounds
WHERE MolWeight > 200;
```

A consulta a seguir exibe todos os registros em que o nome composto ou o peso molecular é nulo:

```
SELECT *
FROM Compounds
WHERE CompoundName IS NULL
OR MolWeight IS NULL;
```

A consulta a seguir exibe todos os registros em que 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;
```

Related reference:

- “Origem de Dados de Documentum – Consultas de Exemplo” na página 188
- “Origem de Dados Excel – Cenário de Amostra” na página 208

Origem de Dados Excel – Cenário de Amostra

Esta seção demonstra uma amostra de implementação do wrapper do Excel₂₀₀₀, acessando uma planilha do Excel 2000 localizada no diretório C:\Data. O cenário registra o wrapper, um servidor e um pseudônimo que será utilizado para acessar a planilha. As instruções mostradas no cenário são digitadas utilizando o Processador da Linha de Comandos do DB2. Depois de registrar o wrapper, você poderá executar consultas na planilha.

O cenário começa com uma planilha composta, denominada *Compound_Master.xls*, com 4 colunas e 9 linhas. O nome do caminho completo para o arquivo é C:\Data\Compound_Master.xls. O conteúdo é mostrado na Tabela 35.

Tabela 35. Planilha de amostra Compound_Master.xls

| | A | V | C | D |
|---|------------|-------------|-----|---------|
| 1 | compound_A | 1.23 | 367 | testado |
| 2 | compound_G | | 210 | |
| 3 | compound_F | 0.000425536 | 174 | testado |

Tabela 35. Planilha de amostra Compound_Master.xls (continuação)

| | A | V | C | D |
|---|------------|---------|------|---------|
| 4 | compound_Y | 1.00256 | | testado |
| 5 | compound_Q | | 1024 | |
| 6 | compound_B | 33.5362 | | |
| 7 | compound_S | 0.96723 | 67 | testado |
| 8 | | | | |
| 9 | compound_O | 1.2 | | testado |

Procedimento:

Para acessar uma planilha do Excel 2000 utilizando o wrapper do Excel:

1. Registre o wrapper do Excel_2000:

```
db2 => CREATE WRAPPER Excel_2000 LIBRARY 'db21sx1s.dll'
      OPTIONS(DB2_FENCED 'N')
```

2. Registre o servidor:

```
db2 => CREATE SERVER biochem_lab WRAPPER Excel_2000
```

3. Registre um pseudônimo que refere-se à planilha do Excel:

```
db2 => CREATE NICKNAME Compound_Master (compound_name VARCHAR(40),
weight FLOAT, mol_count INTEGER, was_tested VARCHAR(20))
FOR biochem_lab
OPTIONS ( FILE_PATH 'C:\Data\Compound_Master.xls')
```

O processo de registro está concluído. A origem de dados do Excel agora faz parte do sistema federado e pode ser utilizado nas consultas SQL.

Os exemplos a seguir mostram as consultas SQL de amostra e os resultados obtidos, utilizando a origem de dados do Excel.

- Amostra de consulta SQL: "Forneça-me todos os dados compostos em que mol_count é maior que 100"

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

Resultado: Todos os campos para as linhas 1, 2, 3, 5 e 7.

- Amostra de consulta SQL: "Forneça-me o compound_name e o mol_count para todos os compostos em que o mol_count ainda não foi determinado.

```
SELECT compound_name, mol_count FROM compound_master
WHERE mol_count IS NULL
```

Resultados: Campos compound_name & mol_count das linhas 4, 6 e 9 da planilha.

- Consulta SQL de amostra: "Contagem do número de compostos que não foram testados e de peso maior que 1."

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

Resultado: A contagem do registro 1 que representa a linha única 6 da planilha que corresponde aos critérios.

- Amostra de consulta SQL: "Forneça-me o compound_name e o mol_count para todos os compostos em que o mol_count foi determinado e é menor que o mol_count médio."

```
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 a média 368 para a consulta principal, que retorna a Tabela 36:

Tabela 36. Resultados da consulta

| COMPOUND_NAME | MOL_COUNT |
|---------------|-----------|
| compound_A | 367 |
| compound_G | 210 |
| compound_F | 174 |
| compound_S | 67 |

Related tasks:

- "Incluindo Excel a um Sistema Federado" na página 205

Related reference:

- "Origem de Dados Excel – Consultas de Exemplo" na página 207

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

O sistema de gerenciamento do banco de dados acessa arquivos do Excel com a autoridade da propriedade LOG ON AS do serviço de banco de dados do DB2. Essa definição pode ser exibida na página de propriedades LOG ON para a instância do DB2. A página de propriedades é acessada através do painel de controle Serviços do Windows NT.

Related reference:

- "Modelo de Controle de Acesso de Arquivo para o Wrapper do Arquivo Estruturado de Tabela" na página 159

- “Controle de Acesso para o Wrapper do Documentum” na página 194

Mensagens para o Wrapper do Excel

Esta seção lista e descreve mensagens que podem ser encontradas enquanto se trabalha com o wrapper do Excel.

Tabela 37. Mensagens emitidas pelo wrapper para o Excel

| Código de Erro | Mensagem | Explicação |
|-----------------------|---|--|
| SQL1817N | A instrução CREATE SERVER não identifica "VERSION" da origem de dados que foi definida para o banco de dados federado. | O parâmetro VERSION não foi especificado durante a instrução CREATE SERVER. Corrija a instrução SQL e execute-a novamente. |
| SQL1822N | Código de erro inesperado "-1000.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro de alocação de memória" | Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1001.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Opção desconhecida". | A opção especificada na instrução DDL não é suportada. Corrija a instrução SQL e execute-a novamente. |
| SQL1822N | Código de erro inesperado "-1002.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Falha na criação do objeto DELTA". | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1100.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Opções de wrappers não suportadas" | OPTIONS de wrappers não são suportadas por este wrapper. Corrija a instrução SQL e execute-a novamente. |

Tabela 37. Mensagens emitidas pelo wrapper para o Excel (continuação)

| Código de Erro | Mensagem | Explicação |
|-----------------------|--|---|
| SQL1822N | Código de erro inesperado "-1200.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "<option> é uma opção de Servidor não suportada" | A opção especificada não é suportada por este wrapper. Corrija a instrução SQL e execute-a novamente. |
| SQL1822N | Código de erro inesperado "-1201.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao obter o nome do servidor" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1209. <código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto tokens associados são "Erro ao converter dados VARCHAR" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1211.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao converter dados INTEGER" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1212.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao converter dados FLOAT" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1400.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "<option> é uma opção de Usuário não suportada" | A opção especificada não é suportada por este wrapper. Corrija a instrução SQL e execute-a novamente. |

Tabela 37. Mensagens emitidas pelo wrapper para o Excel (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|---|
| SQL1822N | Código de erro inesperado "-1401.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Falha na criação do objeto USER Delta" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1500.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "<option> é uma opção de Pseudônimo não suportada" | A opção especificada não é suportada por este wrapper. Corrija a instrução SQL e execute-a novamente. |
| SQL1822N | Código de erro inesperado "-1501.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Opção PATH obrigatória não especificada" | A opção PATH é obrigatória para registrar o NICKNAME. Corrija a instrução SQL e execute-a novamente. |
| SQL1822N | Código de erro inesperado "-1502.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Falha na criação do objeto NICKNAME Delta" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1503.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao obter o tipo de coluna Nickname" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |

Tabela 37. Mensagens emitidas pelo wrapper para o Excel (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL1822N | Código de erro inesperado "-1504.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao obter o nome do tipo de coluna Nickname" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1505.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "recebido da origem de dados do "Wrapper do Excel". | O <tipo de dados> especificado não é suportado por este wrapper. Corrija a instrução SQL e execute-a novamente. |
| SQL1822N | Código de erro inesperado "-1506.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao obter informações da coluna Nickname" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1507.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "A opção <option> não pode ser eliminada" | A opção especificada não pode ser eliminada porque é uma opção obrigatória. |
| SQL1822N | Código de erro inesperado "-1508.VANI" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Os nomes das colunas não podem ser alterados" | A alteração dos nomes das colunas não é permitida pelo wrapper do Excel. |

Tabela 37. Mensagens emitidas pelo wrapper para o Excel (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL1822N | Código de erro inesperado. "-1509.VCTS" recebido da origem de dados "Wrapper Excel". O texto e os tokens associados são "Não foram localizadas informações da coluna". | As informações da coluna não foram localizadas. |
| SQL1822N | Código de erro inesperado "-1701.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao analisar o SQL" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1702.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao acessar o objeto NICKNAME" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1703.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao construir área de armazenamento de dados" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1704.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao ligar o SQL ao Nickname Data" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |

Tabela 37. Mensagens emitidas pelo wrapper para o Excel (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL1822N | Código de erro inesperado "-1705.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Falha na inicialização do aplicativo Excel" | Falha na inicialização do aplicativo Excel. Confirme se o Excel está instalado no sistema e se foi registrado com a versão correta do wrapper. Verifique a propriedade LOG ON AS para a instância DB2 no painel de controle Serviços do Windows NT. O aplicativo Excel será acessado usando essa autoridade. Confirme se esse usuário possui direitos apropriados ou altere essa propriedade para uma conta autorizada, em seguida reinicie o DB2 e execute a consulta SQL novamente. |
| SQL1822N | Código de erro inesperado "-1706.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao abrir as planilhas de origem" | Ocorreu um problema ao abrir a planilha referenciada pelo pseudônimo na consulta SQL. Assegure que o arquivo existe no PATH especificado durante a instrução CREATE NICKNAME no registro. |
| SQL1822N | Código de erro inesperado "-1707.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro ao acessar a área de armazenamento da saída DL" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |
| SQL1822N | Código de erro inesperado "-1708.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Falha no encerramento do aplicativo Excel" | Ocorreu um erro de programa interno. Se esse erro persistir após as consultas repetidas, entre em contato com o Suporte a Software da IBM. |

Tabela 37. Mensagens emitidas pelo wrapper para o Excel (continuação)

| Código de Erro | Mensagem | Explicação |
|-----------------------|--|--|
| SQL1822N | Código de erro inesperado "-1711.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro durante a busca, possível tipo não correspondente de dados/coluna" | Os dados buscados durante a consulta SQL eram de um tipo de dados diferente do tipo de dados especificado durante o registro do pseudônimo. Corrija os dados na planilha de origem ou corrija o tipo de dados registrado no pseudônimo. Se isso não resolver o problema, entre em contato com o Suporte a Software da IBM. |
| SQL1822N | Código de erro inesperado "-1900.<código de programa interno>" recebido da origem de dados do "Wrapper do Excel". Texto e tokens associados são "Erro de alocação de memória" | Ocorreu um erro de programa interno. Contate o Suporte de Software da IBM. |

Related reference:

- "Mensagens para o Wrapper do Arquivo Estruturado de Tabela" na página 160
- "Mensagens para o Wrapper do Documentum" na página 194
- "Mensagens para o Wrapper do BLAST" na página 243
- "Mensagens para o Wrapper do XML" na página 268

Capítulo 15. Configurando o Acesso às Origens de Dados do BLAST

Este capítulo explica o que é o BLAST, como incluir origens de dados do BLAST em seu sistema federado e lista as mensagens de erro associadas ao wrapper do BLAST.

O que É BLAST?

BLAST (Basic Local Alignment Search Tool) é um utilitário mantido pelo National Center for Biotechnology Information (NCBI). O BLAST é utilizado para varrer um banco de dados de seqüência de nucleotídeo ou aminoácido em busca de "acertos". Um acerto do BLAST contém um ou mais pares de segmento de alta contagem (HSPs). Um HSP é um par de fragmentos em seqüência, cujo alinhamento é localmente máximo e cuja contagem de semelhança excede algum valor limite. O NCBI oferece um executável, *blastall*, que é utilizado para realizar pesquisas do BLAST nas origens de dados compatíveis com BLAST, como GenBank e SWISS-PROT.

O wrapper do BLAST suporta todos os cinco tipos de pesquisas BLAST: BLASTn, BLASTp, BLASTx, tBLASTn e tBLASTx. Estes tipos são descritos na Tabela 38.

Tabela 38. Tipos de pesquisa do BLAST suportadas pelo wrapper do BLAST

| Tipo de pesquisa do BLAST | Descrição |
|---------------------------|---|
| BLASTn | Um tipo de pesquisa do BLAST na qual uma seqüência de nucleotídeo é comparada com o conteúdo de um banco de dados de seqüência de nucleotídeo para encontrar seqüências com regiões homólogas para regiões da seqüência original. |
| BLASTp | Um tipo de pesquisa do BLAST na qual uma seqüência de aminoácido é comparada com o conteúdo de um banco de dados de seqüência de aminoácido para encontrar seqüências com regiões homólogas para regiões da seqüência original. |

Tabela 38. Tipos de pesquisa do BLAST suportadas pelo wrapper do BLAST (continuação)

| Tipo de pesquisa do BLAST | Descrição |
|----------------------------------|--|
| BLASTx | Um tipo de pesquisa do BLAST na qual uma seqüência de nucleotídeo é comparada com o conteúdo de um banco de dados de seqüência de aminoácido para encontrar seqüências com regiões homólogas para regiões da seqüência original. A seqüência da consulta é traduzida em todos os seis quadros de leitura e cada uma das seqüências resultantes é utilizada para pesquisar no banco de dados de seqüência. |
| tBLASTn | Um tipo de pesquisa do BLAST na qual uma seqüência de aminoácido é comparada com o conteúdo de um banco de dados de seqüência de nucleotídeo para encontrar seqüências com regiões homólogas para regiões da seqüência original. As seqüências no banco de dados de seqüência são traduzidas em todos os seis quadros de leitura e as seqüências resultantes são pesquisadas para encontrar regiões homólogas para regiões da seqüência de consulta. |
| tBLASTx | Um tipo de pesquisa do BLAST na qual uma seqüência de nucleotídeo é comparada com o conteúdo de um banco de dados de seqüência de nucleotídeo para encontrar seqüências com regiões homólogas para as regiões da seqüência original. Em uma pesquisa tBLASTx, tanto a seqüência de consulta quanto o banco de dados de seqüência são traduzidos em todos os seis quadros de leitura e as seqüências resultantes são comparadas para descobrir regiões homólogas. |

A Figura 5 na página 221 mostra como o BLAST funciona com sistemas federados.

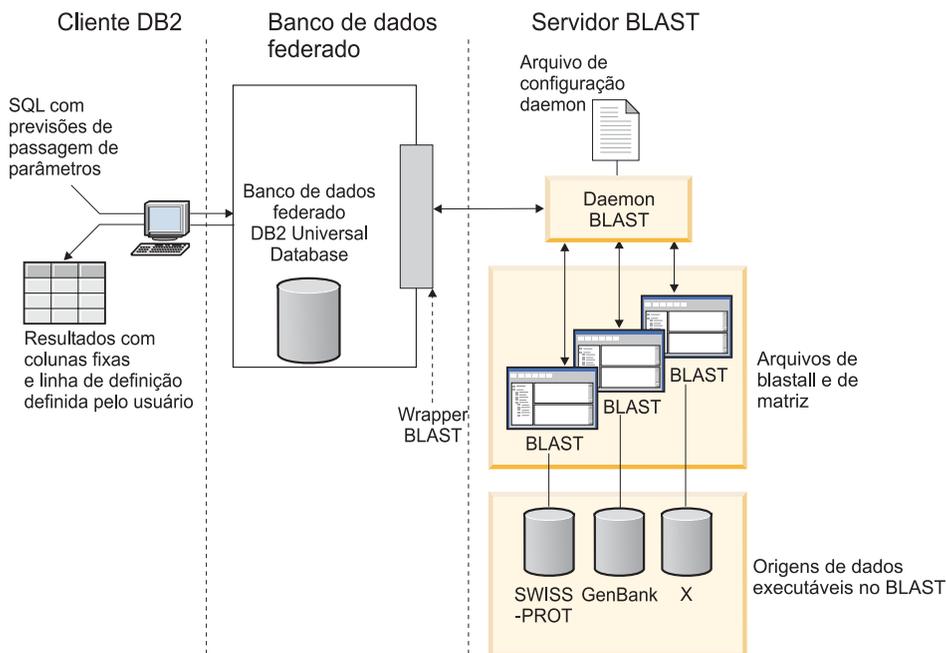


Figura 5. Como o wrapper do BLAST funciona

No lado do cliente, usuários ou aplicativos enviam instruções SQL com predicados de passagem de parâmetros específicos do BLAST que mapeiam por opções BLAST padrão. As instruções SQL com os predicados de entrada são enviados para o seu sistema de banco de dados federado DB2[®] Universal Database com o wrapper do BLAST instalado.

O wrapper do BLAST transforma a consulta em um formato que possa ser entendido pelo aplicativo BLAST e envia a consulta transformada para seu servidor BLAST. Este servidor pode ser uma máquina separada da máquina com o sistema federado. Um programa de daemon especial é executado em seu servidor BLAST. Este daemon, utilizando informações de um arquivo de configuração de daemon, recebe o pedido de consulta do sistema federado e envia para o aplicativo BLAST. O aplicativo BLAST é executado junto a uma origem de dados compatível com BLAST de maneira usual.

Os resultados são retornados para o BLAST e depois para o daemon. O daemon retorna os dados recuperados para o wrapper do BLAST. O wrapper transforma os dados em um formato de tabela relacional e retorna esta tabela para você ou para o aplicativo. Os dados retornados contêm duas partes:

- Uma série de colunas padrão e fixas, familiares para usuários BLAST e
- Informações sobre linha de definição configuradas pelo usuário.

O exemplo a seguir ilustra como as informações relacionais são extraídas das origens de dados compatíveis com BLAST. Os dados mudam do formato de arquivo fasta bruto para um conjunto de dados compatível com BLAST para uma tabela relacional que pode ser unida com outras origens de dados em seu sistema federado.

A Figura 6 é uma amostra de arquivo fasta contendo quatro linhas de definições e registros de seqüência de nucleotídeos.

```
>7:4986 PMON5744
GTTCTTCCCAGTGCCCAAGTCCATTCTGACATCAATGAAGAAGGTAAAAACCTCGCTGATCCCTCTGCC
AAGATGTCGAAATCGACCCGGATAAACTAGCTGCTGTCAAGAATAACAGACAGCCCGGAGGAGATCGTGC
AGAAGTTCCGCAAGGCTGTGACGGACTTCACCTCGGAGGTACCTACGACCCGGCCAGGCAGGAGGCGT
GTCCAATTGGTGGCCATCCACGCGCAGTGACCGGACTCCCGTGGAGGAGGTGGTCCGCCGAAGTCT
GGCATCAACACCCTGGCTACAAGTTGGTGGTGGCGGAGGCTGTGATTGAGAGATTTGCACCAATTAAGA
GTGAAATTGAAAACGAAAGAGGAACAAGGACCACCTAGAGAAGGTTTTACAAGTTGGGTCCGCAAAGC
CAAAGAATTAGCATATCCCGTGTGCCAGGAGGTGAAGAAATTGGTGGGGTTTCTATAGGCAGTCTCACCT
AGTCCCAAGAAAATGTTTTTATCTTGTGGTCTGCTTGCACACTCAGTCTAATAAAGGCAGCTTTCCTAAG
ACGCCAACAAATCCAGTTTGGGGATGCTTAGTTACT
>8:9747 PMON5699
AAGAAGTCTTGTTAGAACCTTCCACCTCCGGCTTCCCCTCCACCTCTCTTACTGTCCCAACCTTCTGAG
ACGCTTTTTCTCTCCCGAGGATTATCTCTTCTCTCTCTCTCTCTCTCTTTTTTTTTTCCCCT
TTCCCCCCCCGAGGCTGGTTTTGCTTTGGGGAGGGGGGTTTTTTAAAGGGCCCGGGGGCCCCCTT
TCCCCCCTAATGGGGTAAATTAATAATGGGGGGGGGGTTTTTTTTTTTAAACCCCTATTTGGTCCGG
CCCGGGGATTTCCCCCCCCCCTTCCCGGTTCCGGGGCCCGAGGAGGGGGGAAAGGGCGGGAA
CCTTTGGTAGTTTCCCCTCGAAAAAATTTTTCGGGGGGAAACCTCCCT
>13:6512 PMON5498
GATAAGAGGCAGAATAGAAGACTGGACTACTTCTCTCCTAAAAACACATTTAAACTAAGCCTGAGCAAT
CTCCACCCAAATGGACCCGAAACCTTAAAAAGAATCCTACTCCTGAAGAAAAGAGGAGGACACATCAA
AGGTAGAAAGGGCGATTTTCATGATATAAACAACCCCATACCTCCAGAGTGGGAAGCTCCACAGACTGAA
AACTAACTGGTTCACAGAACTCACCTACAGGAGTGAGCCCCACATCAAACCTCGAATGTGGGGATCTG
GCACTGGTAGAAAGAGCCCTGGAGCATCTGGCATTGAAGGCCAGTGGGCTTGTGTGCAGGAGATCCAC
AGGACTAGGGGAAACGAGACCCCATTTCTTAAAGGTGCACACAGACTTTTACGTGCACTGGTCCAG
TGCAAAGCAAAGTCTCCATAGGAATCTGGGTCAAACCTGACTGCAGTTCTTGGAGGACCTCCTGGGAAAG
CAAGGGTGAATGTGGCTTCTGTGGGAAAGGACATTGGAAGCAAAGCTCTTGGGAATATTCATCAGTGT
GC
>15:8924 PMON5426
GGAGAACTGACTCCTGAGCAGCTGCAATTCATGCGGCAGGTGCAGCTCGCCAGTGGCAGAAGACGCTG
CCACAGCGCGGACCCGGAACATCGTGACCGCCTGGGCATCGGGCGCTGGTGTGGCAATTTGATCC
GTTTGGACTGTAGACTCAGGGAGACCGCATTAGGGGAACAGGAAGGGCAGCAGGGGCGTGTAGGAGGGC
AGTGTGGGGTGGTAGAAGGAGCCGAGATATGAAAACCTTGGCTCCTTTAACTCTGAATCAAGCGTTT
GGTGTAACCTTACGTTGTCTTTTAAAGGTGATTTTATGATAAATTGATTAATGATTACGGAGTCGGGTGA
GGGCTCCAGGAGCAGACGGCAGAAGATCGAATTTGGGAGGATGATCAGCAGCGTGGTTGAGCAAGTGT
GGGAAAAGGGAATGCGCACATTCACGTGGTTTCTGAACCCACCTCCCAGATGGTTACACCTTCTACT
CGGTGCCAGGAGCGTTTCTTGATGAGCTGGAGGATGAGGCCAAAGCTGCT
```

Figura 6. Amostra de arquivo fasta, nucleo1

O aplicativo formatdb padrão transforma o arquivo fasta em um conjunto de dados compatível com BLAST. Os dados agora estão prontos para serem consultados pelo SQL através de um sistema federado com o wrapper do BLAST instalado e registrado.

A consulta a seguir, enviada por você ou por um aplicativo na extremidade cliente, é transformada pelo wrapper BLAST. Ela é, então, executada junto ao conjunto de dados compatível com BLAST.

```
SELECT Unique_ID, Experiment_Number, Organism_Number, HSP_Info, Score
FROM nucleol
WHERE BlastSeq = 'ACATTCTTATAGAGTATTGCTACTCTCCAGGATAGAGTCATCTCT
GGTCTCCAGAGCCACCGCTGGCTACAAGTTGGTGGTGGCGGAGGCTGTGATTGAGAGATTTG
CACCAATACAGAAACTCACCTACAGGAGTGAGCGGGTGGTAGAAGGAGCCCGAGATATGAAA
ACCTTGTTCAAGACCCATTGTACCGGGG';
```

Os resultados da consulta são transformados pelo wrapper do BLAST em um formato de tabela relacional mostrado na Tabela 39.

Tabela 39. O BLAST retorna resultados em um formulário de tabela relacional quando integrado em seu sistema federado

| ID exclusivo | Número da experiência | Número do organismo | HSP_INFO | CONTAGEM |
|--------------|-----------------------|---------------------|--|------------------------|
| PMON5744 | 4986 | 7 | Identidades = 57/201 (28%), Positivos = 57/201 (28%), Intervalos = 0/201 (0%) | +1.13487000000000E+002 |
| PMON5426 | 8924 | 15 | Identidades = 35/201 (17%), Positivos = 35/201 (17%), Intervalos = 0/201 (0%) | +6.98754000000000E+001 |
| PMON5498 | 6512 | 13 | Identidades = 26/201 (13%), Positivos = 26/201 (13%), Intervalos = 0/201 (0%) | +5.20342000000000E+001 |

Os dados estão em um formulário relacional completo e podem ser unidos com dados de outras origens de dados utilizadas por seu laboratório. Combinar os resultados de várias origens de dados pode levar a descobertas não encontradas facilmente ou eficientemente antes da implementação de seu sistema federado.

Related concepts:

- “O que São Arquivos com Estrutura de Tabela?” na página 151
- “O que É Documentum?” na página 165
- “O que É Excel?” na página 203
- “O que É XML?” na página 247

Related tasks:

- “Incluindo BLAST a um Sistema Federado” na página 224

Incluindo BLAST a um Sistema Federado

Procedimento:

Para incluir a origem de dados do BLAST em um servidor federado:

1. Verifique se a versão correta do executável blastall e arquivos matriz estão instalados.
2. Configure o daemon BLAST. Configure o daemon BLAST.
3. Inicie o daemon BLAST. Inicie o daemon BLAST.
4. Registre o wrapper utilizando a instrução CREATE WRAPPER. Registre o wrapper utilizando a instrução CREATE WRAPPER.
5. Opcional: Defina a variável de ambiente DB2_DJ_COMM para melhorar o desempenho da consulta.
6. Registre o servidor utilizando a instrução CREATE SERVER. Registre o servidor utilizando a instrução CREATE SERVER.
7. Registre pseudônimos utilizando a instrução CREATE NICKNAME. Registre pseudônimos utilizando a instrução CREATE NICKNAME.

As instruções podem ser executadas a partir do processador da linha de comandos do DB2. Depois que o wrapper BLAST for adicionado ao sistema federado, será possível executar consultas na origem de dados BLAST.

Related tasks:

- “Verificando se a Versão Correta do Executável Blastall e os Arquivos de Matriz Estão Instalados” na página 225
- “Configurando o Daemon do BLAST” na página 225
- “Iniciando o Daemon do BLAST” na página 229
- “Registrando o Wrapper do BLAST” na página 230
- “Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper BLAST” na página 230
- “Registrando um Servidor para uma Origem de Dados do BLAST” na página 231

- “Registrando Pseudônimos para Origens de Dados do BLAST” na página 232

Verificando se a Versão Correta do Executável Blastall e os Arquivos de Matriz Estão Instalados

Verificar se a versão correta do executável blastall e dos arquivos de matriz está instalada faz parte da maior tarefa de adicionar BLAST a um sistema federado.

Pré-requisitos:

Verifique se você possui a versão mais recente do executável blastall e os arquivos de matriz BLOSUM62, BLOSUM80, PAM30 e PAM70 em sua máquina de servidor BLAST. Se não possuir, será necessário instalar os arquivos binários e os arquivos de matriz. Os arquivos de matriz devem estar no mesmo diretório do executável blastall.

Procedimento:

Para verificar o nível da versão de seu executável blastall e dos arquivos de matriz:

1. Execute uma pesquisa BLAST a partir da linha de comandos e anote o número de versão localizado no arquivo de saída.
2. Verifique o Web site deste produto para obter versões do BLAST que foram testadas com este wrapper para assegurar que você possui uma versão suportada.

A próxima tarefa nesta seqüência de tarefas é configurar o daemon BLAST.

Related tasks:

- “Configurando o Daemon do BLAST” na página 225

Configurando o Daemon do BLAST

A configuração do daemon BLAST faz parte da maior tarefa de adicionar o BLAST a um sistema federado.

O wrapper do BLAST requer que um daemon do BLAST esteja sendo executado em sua máquina baseada em UNIX acessível através de TCP/IP a partir de seu sistema federado do DB2 Universal Database. O daemon é executado separadamente do wrapper e do DB2 Universal Database e escuta

um pedido de job do BLAST a partir do wrapper. O arquivo executável do daemon, `db2blast_daemon`, pode estar localizado em qualquer diretório na máquina do servidor BLAST.

Durante a instalação do DB2 Universal Database, o executável do daemon é colocado no diretório `/usr/opt/db2_08_01/bin` no AIX e no diretório `/opt/IBM/db2/V8.1/bin` nas outras plataformas Unix, da máquina onde o servidor federado está sendo instalado. Se, em seu ambiente, o BLAST for executado em uma máquina diferente, você deve fazer uma cópia do daemon para uma localização de sua escolha na máquina.

O daemon do BLAST deve ter:

- Acesso de execução ao arquivo binário `blastall` para que ele possa executar pesquisas BLAST.
- Acesso de gravação a um diretório no qual ele possa gravar arquivos temporários.
- Acesso de leitura a, pelo menos, uma origem de dados compatível com BLAST, na qual pesquisas do BLAST podem ser executadas. O executável `blastall` deve ter acesso de leitura ao arquivo de dados e aos arquivos de índice BLAST gerados pelo programa `formatdb`.

O daemon do BLAST requer um arquivo de configuração. Uma amostra de arquivo de configuração do daemon, denominada `BLAST_DAEMON.config`, é colocada no diretório `DB2PATH/samples/lifesci`, em que `DB2PATH` é o diretório no qual o DB2 Universal Database está instalado. `BLAST_DAEMON.config` é o nome padrão para o arquivo.

Copie o arquivo de configuração para qualquer localização acessível ao daemon, renomeie-o se quiser e edite-o para trabalhar com sua origem de dados. Por padrão, o `blast_daemon` procura suas informações de configuração no diretório de trabalho a partir do qual foi iniciado.

Procedimento:

Para configurar o daemon, especifique as seguintes opções no arquivo de configuração. Para opções que requerem caminhos, você pode especificar caminhos relativos. Caminhos relativos são relativos ao diretório a partir do qual o processo do daemon foi iniciado.

DAEMON_PORT

Esta é a porta de rede na qual o daemon escutará pedidos de job do BLAST enviados pelo wrapper.

MAX_PENDING_REQUESTS

Este é o número máximo de pedidos de job do BLAST que podem estar bloqueando o daemon a qualquer hora. Este número não

representa o número de jobs do BLAST que estão sendo executados simultaneamente, apenas o número de pedidos de job que podem bloquear de uma vez. Recomendamos que você defina este valor para um número maior que cinco. O daemon do BLAST não restringe o número de jobs do BLAST que podem ser executados simultaneamente.

DAEMON_LOGFILE_DIR

Este é o diretório no qual o daemon criará seu arquivo de log. Este arquivo conterá informações úteis de status e de erro geradas pelo daemon do BLAST.

Q_SEQ_DIR_PATH

Este é o diretório no qual um arquivo temporário de dados de seqüência de consulta será criado pelo daemon. Este arquivo temporário é excluído quando o job do BLAST é concluído.

BLAST_OUT_DIR_PATH

Este é o diretório no qual o daemon criará o arquivo temporário para armazenar os dados de saída do BLAST. Os dados serão lidos a partir deste arquivo e passados de volta para o wrapper através da conexão de rede, neste ponto o daemon elimina o arquivo temporário.

BLASTALL_PATH

Este é o nome completo do arquivo executável do BLAST na máquina que executa o daemon.

entrada de especificação do banco de dados

Especifica a localização de uma origem de dados compatível com BLAST. Para que o daemon funcione corretamente, é necessário especificar cada nome de entrada utilizado no arquivo de configuração na opção DATASOURCE da instrução CREATE NICKNAME quando você criar o pseudônimo para a origem de dados.

O arquivo de configuração deve conter pelo menos uma entrada de especificação de banco de dados no seguinte formato:

```
entry_name = path to BLAST-able_data_source
```

Por exemplo, para especificar a origem de dados compatível com BLAST GenBank, você poderia incluir a seguinte linha ao arquivo de configuração do daemon:

```
genbank=/dsk/1/nuc1_data/genbank
```

O caminho indicado em uma entrada de especificação de banco de dados deve conter os três arquivos de índice.

- Para origens de dados de nucleotídeos, os arquivos de índice possuem estas extensões:

- .nhr
- .nin
- .nsq
- Para origens de dados de aminoácidos, os arquivos de índice possuem estas extensões:
 - .phr
 - .pin
 - .psq

As entradas de especificação de banco de dados devem indicar o nome do arquivo que contém os dados formatados como Fasta originais. Os três arquivos de índice devem ter o mesmo nome de arquivo raiz contendo os dados formatados como Fasta originais.

A primeira linha no arquivo de configuração deve ser um sinal de igual. Se o sinal de igual estiver ausente, o daemon não será inicializado. Uma mensagem de erro indicará que DAEMON_PORT não foi especificado.

A última linha do arquivo de configuração deve ser finalizada por uma nova linha. Se não for, você receberá uma mensagem de erro quando tentar executar a primeira consulta do BLAST utilizando a origem de dados listada na última linha. O arquivo de configuração de amostra fornecido não possui a última linha finalizada por uma nova linha. Para que ele seja executado corretamente, será necessário finalizar a última linha com uma nova linha.

Exemplo:

O exemplo a seguir mostra o conteúdo de uma amostra de arquivo de configuração, com as opções requeridas e especificações de origem de dados compatíveis com BLAST para GenBank e SWISS-PROT.

```
=
DAEMON_PORT=4007
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=./
Q_SEQ_DIR_PATH=./
BLAST_OUT_DIR_PATH=./
BLASTALL_PATH=./blastall
genbank=/dsk/1/nuc1_data/genbank
swissprot=/dsk/1/prot_data/swissprot
```

A próxima tarefa nesta seqüência de tarefas é iniciar o daemon BLAST.

Related tasks:

- “Iniciando o Daemon do BLAST” na página 229

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper BLAST” na página 358

Iniciando o Daemon do BLAST

O início do daemon BLAST faz parte da maior tarefa de adicionar BLAST a um sistema federado. Para você acessar origens de dados do BLAST, é necessário que o daemon do BLAST esteja sendo executado.

Pré-requisitos:

Antes de iniciar o daemon BLAST, você deve ter acesso de gravação a todos os caminhos listados nas entradas DAEMON_LOGFILE_DIR, BLAST_OUT_DIR_PATH e Q_SEQ_DIR_PATH no arquivo de configuração.

Procedimento:

Para iniciar o daemon do BLAST se estiver no diretório de instalação do daemon, não alterou o nome do arquivo de configuração do daemon e o arquivo de configuração estiver no mesmo diretório do arquivo executável do daemon, digite o seguinte comando na linha de comandos:

```
db2blast_daemon
```

O executável inicia um novo processo no qual o daemon do BLAST é executado.

Para iniciar o daemon do BLAST se você tiver alterado o nome do arquivo de configuração do daemon ou se não estiver no diretório no qual o arquivo de configuração do daemon está localizado, você deve utilizar a opção -c no comando do daemon do wrapper para apontar o executável do daemon para o novo nome ou localização.

Por exemplo, o comando a seguir faz com que o daemon do wrapper procure suas informações de configuração em um arquivo denominado BLAST_D.config no subdiretório cfg.

```
db2blast_daemon -c cfg/BLAST_D.config
```

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper BLAST.

Related tasks:

- “Registrando o Wrapper do BLAST” na página 230

Registrando o Wrapper do BLAST

O registro do wrapper BLAST faz parte da maior tarefa de adicionar BLAST a um sistema federado. Você deve registrar o wrapper para acessar uma origem de dados. Wrappers são mecanismos que os servidores federados utilizam para se comunicar com as origens de dados e recuperar dados delas. Os wrappers são instalados no sistema como arquivos de biblioteca.

Procedimento:

Para registrar o wrapper do BLAST, envie a instrução CREATE WRAPPER.

Por exemplo, para criar um wrapper do BLAST no AIX denominado `my_blast`, a partir do arquivo de biblioteca padrão, `libdb21sblast.a`, envie a seguinte instrução:

```
CREATE WRAPPER my_blast LIBRARY 'libdb21sblast.a'  
  OPTIONS(DB2_FENCED 'N');
```

Para obter uma tabela de nomes de arquivos de biblioteca padrão para o wrapper BLAST por plataforma suportada, consulte a seção de tarefas relacionadas.

A próxima tarefa nesta seqüência de tarefas é definir a variável de ambiente `DB2_DJ_COMM` para o wrapper BLAST.

Related tasks:

- “Definindo a Variável de Perfil `DB2_DJ_COMM` do DB2 para o Wrapper BLAST” na página 230
- “Depois de Instalar Wrappers Não Relacionais” no *DB2 Information Integrator Installation Guide*

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Definindo a Variável de Perfil `DB2_DJ_COMM` do DB2 para o Wrapper BLAST

A definição da variável de perfil `DB2_DJ_COMM` do DB2 para o wrapper BLAST faz parte da maior tarefa de adicionar BLAST a um sistema federado. Para aprimorar o desempenho quando origens de dados BLAST são acessadas, defina a variável de perfil `DB2_DJ_COMM` do DB2. Esta variável determina se o servidor federado carregará o wrapper na inicialização.

Procedimento:

Para definir a variável de perfil DB2_DJ_COMM do DB2, envie o comando db2set com a biblioteca do wrapper que corresponde ao wrapper especificado na instrução CREATE WRAPPER associada.

Por exemplo:

```
db2set DB2_DJ_COMM='libdb21sblast.a'
```

Verifique se não existem espaços em nenhum lado do sinal de igual (=).

Existe overhead associado ao carregamento das bibliotecas do wrapper durante a inicialização do banco de dados. Para evitar este overhead, especifique apenas as bibliotecas que você pretende acessar.

A próxima tarefa nesta seqüência de tarefas é registrar o servidor para uma origem de dados BLAST.

Related tasks:

- “Registrando um Servidor para uma Origem de Dados do BLAST” na página 231

Registrando um Servidor para uma Origem de Dados do BLAST

O registro do servidor para uma origem de dados BLAST faz parte da maior tarefa de adicionar BLAST a um sistema federado. Depois de registrar o wrapper, você deve registrar um servidor correspondente.

Procedimento:

Para registrar o servidor do BLAST no sistema federado, utilize a instrução CREATE SERVER.

Para cada máquina na qual o executável e o daemon do BLAST estiverem instalados em seu ambiente, você deve registrar um servidor para cada tipo de pesquisa do BLAST que quiser executar utilizando esta instância do executável e do daemon do BLAST.

Por exemplo, para registrar um servidor chamado blast_server1 para o wrapper my_blast criado utilizando a instrução CREATE WRAPPER, que será usada para pesquisas BLASTn, envie a seguinte instrução:

```
CREATE SERVER blast_server1
  TYPE blastn
  VERSION 2.1.2
  WRAPPER my_blast
  OPTIONS (NODE 'big_rs.company.com', DAEMON_PORT '4007')
```

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para origens de dados BLAST.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados do BLAST” na página 232

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*
- “Argumentos da Instrução CREATE SERVER - Wrapper BLAST” na página 380

Registrando Pseudônimos para Origens de Dados do BLAST

O registro de pseudônimos para origens de dados BLAST faz parte da maior tarefa de adicionar um BLAST a um sistema federado. Depois de registrar o servidor, você deve registrar um pseudônimo correspondente. Pseudônimos são utilizados quando você faz referência a uma origem de dados do BLAST em uma consulta.

Procedimento:

Para registrar um pseudônimo do BLAST, utilize a instrução CREATE NICKNAME. . Como cada tipo de pesquisa do BLAST é tratado por um servidor diferente, você deve definir um pseudônimo separado para cada tipo de pesquisa do BLAST que deseja executar em uma determinada origem de dados compatível com BLAST.

O pseudônimo especifica informações sobre coluna para a parte de linha de definição da origem de dados. Todas as outras colunas são fixas. Para obter mais informações sobre a análise de linha de definição, consulte “Análise de Linha de Definição”. Para obter mais informações sobre colunas fixas, consulte “Colunas Fixas” na página 233.

Análise de Linha de Definição

A linha de definição, também denominada defline, é como uma chave para cada seqüência na origem de dados compatível com BLAST e é retornada como parte de cada acerto do BLAST.

Se você estiver interessado em incluir as informações sobre a linha de definição em sua tabela de resultados, você deve especificar as colunas da linha de definição na instrução CREATE NICKNAME. Cada especificação de coluna deve especificar uma opção INDEX. A opção DELIMITER deve ser especificada para cada coluna, exceto para a última coluna especificada se você quiser que esta coluna contenha o restante da linha de definição.

Os campos de linha de definição devem ser do tipo integer, float, double ou varchar.

Se dados forem encontrados no campo Accession Number de um acerto do BLAST, estes dados são inseridos antes dos dados no campo Definition do acerto do BLAST. A linha de definição resultante que contém os dados Accession Number seguidos pelos dados do campo Definition é analisada pelo wrapper.

Colunas Fixas

A instrução CREATE NICKNAME cria automaticamente colunas fixas. As colunas fixas não aparecem na instrução CREATE NICKNAME, mas fazem parte da definição do pseudônimo e podem ter referência em consultas SQL. Há dois tipos de colunas fixas, de entrada e de saída.

Colunas fixas de entrada

Colunas fixas de entrada são utilizadas como predicados de passagem de parâmetro em consultas SQL. Elas passam os switches BLAST padrão para o BLAST. O BLAST é então executado na origem de dados especificada utilizando estas chaves. Colunas fixas de entrada também podem ter referência na lista de seleção de consulta e podem ser retornadas como parte da tabela de resultado. Colunas fixas de entrada são listadas na Tabela 40.

Tabela 40. Colunas fixas de entrada

| Nome | Tipo de dados | Operadores permitidos | Descrição |
|--------------|----------------|-----------------------|---|
| BlastSeq | varchar(32000) | = | Passa a seqüência de consulta para o wrapper do BLAST. |
| E_Value | double | < | Parâmetro de entrada e de saída. Como um parâmetro de entrada, esta coluna indica para o wrapper do BLAST o limite superior de valores de espera que devem ser retornados do blastall. |
| QueryStrands | integer | = | Especifica quais margens devem ser comparadas quando uma pesquisa BLASTn estiver sendo realizada. Um valor igual a 1 indica que a margem superior deve ser utilizada, 2 indica a margem inferior e 3 indica que as duas margens devem ser comparadas. |

Tabela 40. Colunas fixas de entrada (continuação)

| Nome | Tipo de dados | Operadores permitidos | Descrição |
|------------------|---------------|-----------------------|--|
| GapAlign | char(1) | = | Indica ao wrapper se alinhamentos com intervalo são permitidos na saída do BLAST. |
| Matrix | varchar(50) | = | Determina qual matriz de substituição é utilizada pelo blastall para determinar o grau de semelhança entre pares de aminoácidos. Apenas os tipos de pesquisa do BLAST que comparam aminoácidos a aminoácidos utilizam este predicado. |
| NMisMatchPenalty | integer | = | Especifica o valor que o blastall deduz do score de um alinhamento se um dos pares de nucleotídeos na região homóloga não corresponder. Apenas os tipos de pesquisa do BLAST que comparam nucleotídeos a nucleotídeos utilizam este predicado. |
| NMatchReward | integer | = | Especifica o valor que o blastall inclui ao score de um alinhamento para cada um dos pares de nucleotídeos na região homóloga que corresponder. Apenas os tipos de pesquisa do BLAST que comparam nucleotídeos a nucleotídeos utilizam este predicado. |
| FilterSequence | char(1) | = | Indica ao blastall se ele deve realizar filtragem para remover segmentos desinteressantes biologicamente da seqüência de consulta. Se o tipo de consulta for BLASTn, o filtro utilizado é DUST. De outro modo, a filtragem é realizada por SEG. |

Tabela 40. Colunas fixas de entrada (continuação)

| Nome | Tipo de dados | Operadores permitidos | Descrição |
|--------------------|---------------|-----------------------|---|
| NumberOfAlignments | integer | = | Especifica quantos alinhamentos HSP incluir em uma saída BLAST. |
| GapCost | integer | = | Especifica o valor que o blastall deduz do score de um alinhamento se um intervalo tiver que ser introduzido na seqüência de consulta ou na seqüência de acertos para permitir que o comprimento do alinhamento cresça. |
| ExtendedGapCost | integer | = | Especifica o valor que o blastall deduz do score de um alinhamento se um intervalo que já foi introduzido na seqüência de consulta ou na seqüência de acertos tiver que ser estendido por um nucleotídeo ou aminoácido para permitir que o comprimento do alinhamento cresça. |
| WordSize | integer | = | Indica ao blastall o comprimento dos acertos iniciais que o blastall pesquisa inicialmente no banco de dados. |
| ThresholdEx | integer | = | Indica o limite de score abaixo do qual o BLAST não tenta estender mais um acerto. |

Os tipos de pesquisa e switches do BLAST suportados para cada coluna fixa de entrada são listados na Tabela 41 na página 236

Tabela 41. Tipos de pesquisa e switches do BLAST suportados pelas colunas fixas de entrada

| Nome | Tipos de pesquisa do BLAST | Switch do BLAST | Req? | Padrão |
|---|----------------------------|-----------------|------|-------------------------|
| BlastSeq | n, p, x, tn, tx | -l | S | N/C |
| E_Value | n, p, x, tn, tx | -e | N | 10 |
| QueryStrands | n | S | N | 3 |
| GapAlign | n, p, x, tn, tx | -g | N | T |
| Matrix | p, x, tn, tx | -n | N | BLOSUM62 |
| NMismatchPenalty | n | -q | N | -3 |
| NMatchReward | n | -r | N | 1 |
| FilterSequence | n, p, x, tn, tx | -F | N | T |
| NumberOfAlignments | n, p, x, tn, tx | -b | N | 250 |
| GapCost | n, p, x, tn, tx | -G | N | 11 |
| ExtendedGapCost | n, p, x, tn, tx | -E | N | 1 |
| WordSize (para Blastn, um valor menor que 7 é inválido) | n, p, x, tn, tx | -W | N | 11 -BLASTn 3 -BLASTp |
| ThresholdEx | n, p, x, tn, tx | -f | N | 0 |

Colunas fixas de saída

Colunas fixas de saída são retornadas na tabela de resultados da consulta e podem ser utilizadas como predicados. Colunas fixas de saída são listadas na Tabela 42.

Tabela 42. Colunas fixas de saída

| Nome | Tipo de dados | Descrição |
|-------|---------------|--|
| Score | double | O score computado para um HSP, conforme reportado nos resultados do BLAST. |

Tabela 42. Colunas fixas de saída (continuação)

| Nome | Tipo de dados | Descrição |
|----------------------|---------------|---|
| E_value | double | Parâmetro de entrada e de saída. Como um parâmetro de saída, esta coluna fornece o score computado para um HSP, conforme reportado nos resultados do BLAST. |
| Tamanho | integer | O comprimento da sequência de acertos, conforme reportado nos resultados do BLAST. |
| HSP_Info | varchar(100) | A cadeia de informações para o HSP dado, conforme reportado pelo BLAST. Esta cadeia contém informações sobre o número de nucleotídeos ou aminoácidos que corresponderam entre a sequência de consulta e a sequência de acertos. |
| HSP_ALIGNMENT_LENGTH | integer | O comprimento do alinhamento HSP. |
| HSP_IDENTITY | integer | A identidade de porcentagem de alinhamento definida como o número de identidades dividido pelo comprimento do alinhamento. |
| HSP_GAPS | integer | Os intervalos de porcentagem no alinhamento definidos como o número de intervalos dividido pelo comprimento do alinhamento. |
| HSP_POSITIVE | integer | Os positivos de porcentagem de alinhamento definidos como o número de positivos dividido pelo comprimento do alinhamento. |
| HSP_QUERY_FRAME | integer | O quadro de leitura do alinhamento na sequência de consultas. Disponível apenas para os tipos de servidores blastx, tblastn e tblastx. |
| HSP_HIT_FRAME | integer | O quadro de leitura do alinhamento na sequência de acertos. Disponível apenas para os tipos de servidores blastx, tblastn e tblastx. |
| HSP_Q_Start | integer | A posição numérica do primeiro nucleotídeo ou aminoácido homólogo na sequência de consulta. |

Tabela 42. Colunas fixas de saída (continuação)

| Nome | Tipo de dados | Descrição |
|-------------|----------------|--|
| HSP_Q_End | integer | A posição numérica do último nucleotídeo ou aminoácido homólogo na seqüência de consulta. |
| HSP_Q_Seq | varchar(32000) | O segmento da seqüência de consulta começando com HSP_Q_Start e terminando com HSP_Q_End. |
| HSP_H_Start | integer | A posição numérica do primeiro nucleotídeo ou aminoácido homólogo na seqüência de acertos. |
| HSP_H_End | integer | A posição numérica do último nucleotídeo ou aminoácido homólogo na seqüência de acertos. |
| HSP_H_Seq | varchar(32000) | O segmento da seqüência de acertos começando com HSP_H_Start e terminando com HSP_H_End. |
| HSP_Midline | varchar(32000) | A saída de cadeia pelo BLAST que indica o grau de homologia entre os aminoácidos e os nucleotídeos em cada posição nas regiões homólogas das seqüências de consulta e de acerto. |

Não há mais tarefas nesta seqüência de tarefas.

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper BLAST” na página 358
- “Instrução CREATE NICKNAME - Exemplos para Wrapper BLAST” na página 238

Instrução CREATE NICKNAME - Exemplos para Wrapper BLAST

A instrução CREATE NICKNAME a seguir define o pseudônimo genbank.

Ela assume que o campo de definição em um resultado BLAST contém as seguintes informações:

```
>276342 15:8924 PMON5426
```

em que:

276342 O campo de ascensão do resultado do BLAST.

15:8924 PMON5426

O campo de definição em um resultado do BLAST contendo um número de organismo seguido por um número de experimento e por um identificador exclusivo.

Com estas informações, o pseudônimo a seguir é criado:

```
CREATE NICKNAME genbank (  
  acc_num integer  OPTIONS(INDEX '1', DELIMITER ' '),  
  org_num integer  OPTIONS(INDEX '2', DELIMITER ':'),  
  exp_num integer  OPTIONS(INDEX '3', DELIMITER ' '),  
  u_id varchar(10) OPTIONS(INDEX '4')  
  FOR SERVER blast_server1  
  OPTIONS(DATASOURCE 'genbank', TIMEOUT '300');
```

A coluna `acc_num` poderia conter 276342, a coluna `org_num` poderia conter 15, a coluna `exp_num` poderia conter 8924 e a coluna `u_id` poderia conter PMON5426.

Depois de enviar a instrução `CREATE NICKNAME`, você pode utilizar o pseudônimo `genbank` para consultar o sistema federado. Pode também unir o pseudônimo `genbank` a outros pseudônimos e tabelas em seu sistema federado.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados do BLAST” na página 232

Related reference:

- “Sintaxe da Instrução `CREATE NICKNAME` - Wrapper BLAST” na página 358

Configurando o TurboBlast para Funcionar com o Wrapper BLAST

Restrições:

O TurboBlast não suporta algumas opções de comando `blastall`. Por exemplo, a opção de alinhamento com intervalo `-g F` não é suportada. Se você especificar `F` para o valor da coluna `GapAlign` em seu pseudônimo BLAST, o TurboBlast gerará um erro. Para obter uma lista completa de opções não suportadas, consulte o *TurboBlast 2.0 User Guide*.

Procedimento:

Para configurar o TurboBlast para funcionar com o wrapper BLAST:

1. Instale e configure o wrapper BLAST. Execute uma consulta em um banco de dados compatível com o `blast` para testar a instalação.

2. O wrapper BLAST e o TurboBlast suportam as plataformas AIX, Linux, Solaris e Windows NT/2000. O daemon BLAST não está disponível nos sistemas operacionais Windows NT/2000. O daemon funcionará com o TurboBlast no Windows NT/2000 quando o daemon BLAST estiver disponível nestes sistemas operacionais.
3. Instale e configure o TurboBlast de acordo com o *TurboBlast 2.0 Installation and Reference Guide*. Você pode instalar e configurar o sistema TurboBlast de várias formas. Para permitir que o wrapper BLAST funcione com o TurboBlast, é necessário instalar e configurar o TurboBlast Client na máquina em que está localizado o daemon BLAST. O daemon BLAST pode chamar o comando `tblastall`.
4. Certifique-se de testar o sistema TurboBlast depois de instalar e configurar o TurboBlast. Siga as instruções no *TurboBlast 2.0 Installation and Reference Guide*.
5. Altere o arquivo `BLAST_DAEMON.config` conforme a seguir:
 - a. Especifique o parâmetro `BLASTALL_PATH` como o caminho completo do `tblastall`. Por exemplo:
`BLASTALL_PATH=/home/blasttst/turboblast/TBlast-2.1/tblastall`
 - b. Especifique a entrada de especificação do banco de dados compatível com blast como o nome do banco de dados compatível com blast utilizado para fazer upload do banco de dados compatível com blast para o TurboBlast. Os nomes dos bancos de dados são mostrados quando você insere o comando `listdatabase -l` no TurboBlast. Este nome do banco de dados TurboBlast deve ser utilizado em vez do caminho para a origem de dados compatível com blast. Por exemplo:
`genbank=<the genbank database name in TurboBlast>`
6. Inicie novamente o daemon BLAST. O daemon blast chama `tblastall` em vez de `blastall` para fazer trabalho de pesquisa em bancos de dados compatíveis com blast.
7. Os arquivos de log relacionados a `tblastall` são gravados no `DAEMON_LOGFILE_DIR` especificado no arquivo `BLAST_DEAMON.config`. Verifique também `STDERR.log` e `STDOUT.log` produzidos pelo daemon blast no mesmo diretório.

Construindo Consultas SQL BLAST

SQL para origens de dados BLAST deve conter apenas predicados de entrada especiais utilizados para passar switches BLAST padrão para o arquivo executável `blastall`.

Restrições:

Para ser válida, cada consulta passada para o wrapper do BLAST deve conter pelo menos o predicado de entrada BlastSeq. Todos os outros predicados são opcionais.

Procedimento:

Para construir uma consulta BLAST, utilize os predicados de entrada na cláusula WHERE de sua instrução SQL.

O exemplo a seguir mostra três predicados de entrada: BlastSeq, GapCost e NMismatchPenalty.

```
Select * from blast b where  
BlastSeq = 'GTCCAGCC...' AND  
GapCost = -10 AND  
NMismatchPenalty = -4;
```

Related tasks:

- “Registrando Pseudônimos para Origens de Dados do BLAST” na página 232

Related reference:

- “Origem de Dados BLAST – Consultas de Exemplo” na página 241

Origem de Dados BLAST – Consultas de Exemplo

Várias consultas de amostra do BLAST são fornecidas para ilustrar como as consultas são construídas para origens de dados BLAST.

Para executar consultas, utilize os exemplos como guia.

Nestas consultas, o nome utilizado para cada pseudônimo indica o tipo de pesquisa do BLAST e a origem de dados. Isto é feito para que as instruções de registro não precisem ser listadas com cada amostra de consulta. Além disso, algumas das consultas utilizam outras origens de dados hipotéticas para que estes exemplos possam ilustrar o comportamento do wrapper quando é unido com outras origens de dados.

Consulta 1

```
select *  
from blastn_genbank  
where BlastSeq =  
'caaccctccagcgagttgtcaatggcgaggaagctgttccccac';
```

Quando esta instrução SQL for executada, o wrapper realizará uma pesquisa BLASTn do GenBank utilizando a seqüência indicada. O wrapper retornará todas as colunas disponíveis, incluindo as colunas de parâmetros de entrada e as colunas de resultados do BLAST.

Consulta 2

```
select *
from blastn_genbank
where BlastSeq =
  'caaccctccagccgagttgtcaatggcgaggaagctgttccccac'
and GapCost = 8 and NmisMatchPenalty = -4;
```

Quando esta instrução SQL for executada, o wrapper realizará uma pesquisa BLASTn do GenBank utilizando a seqüência indicada. Além disso, o wrapper passará os dois parâmetros indicados para o daemon e eles serão passados para a linha de comandos blastall. O wrapper retornará todas as colunas disponíveis, incluindo as colunas de parâmetros de entrada e as colunas de resultados do BLAST.

Consulta 3

```
select blp.*
from blastp_swissprot blp, protein_db prdb
where prdb.keyword = 'malic enzyme'
and blp.BlastSeq = prdb.sequence;
```

Quando esta instrução SQL for executada, o wrapper realizará zero ou mais pesquisas BLASTp do SWISS-PROT, dependendo do número de seqüências retornadas de um banco de dados de seqüência de proteínas hipotético. Esta instrução será dividida em duas consultas separadas pelo DB2 e uma pesquisa BLASTp será executada para cada linha que for retornada do banco de dados de proteínas hipotético. O wrapper retornará todas as colunas disponíveis, incluindo as colunas de parâmetros de entrada e as colunas de resultados do BLAST.

Consulta 4

```
select Score, E_Value, HSP_Info, HSP_Q_Seq, HSP_H_Seq, HSP_Midline
from blastx_swissprot
where BlastSeq = 'gagttgtcaatggcgagg'
and GapCost = 8;
```

Quando esta instrução SQL for executada, o wrapper realizará uma pesquisa BLASTx no SWISS-PROT utilizando a seqüência indicada. Neste caso, o blastall traduzirá a seqüência de entrada em todos os seis quadros de leitura e realizará a pesquisa de homologia utilizando cada uma das seis novas seqüências de proteínas criadas. Os HSPs nos resultados conterão alinhamentos aminoácido-aminoácido, ao invés de alinhamentos nucleotídeo-nucleotídeo. O parâmetro fornecido será passado para o daemon

e depois para o blastball através da linha de comandos. O wrapper retornará apenas aquelas colunas que forem especificamente solicitadas na consulta.

Consulta 5

```
select tblx.Score, tblx.E_Value, tblx.HSP_Info tblx.HSP_Q_Seq,  
       HSP_H_Seq, HSP_Midline  
from tblastx_genbank tblx, gen_exp_database gedb  
where tblx.BlastSeq = gedb.sequence  
       and gedb.organism = 'interesting organism'  
       and GapCost = 8  
       and FilterSequence = 'F';
```

Quando esta instrução SQL for executada, o wrapper realizará zero ou mais pesquisas tBLASTx do GenBank, dependendo do número de seqüências retornadas de um banco de dados de expressão de genes hipotético. A instrução será dividida em duas consultas separadas pelo DB2 e uma pesquisa tBLASTx será executada para cada linha que for retornada do banco de dados de expressão de gene hipotético. Neste caso, o blastall traduzirá a seqüência de entrada e todas as seqüências do GenBank em todos os seis quadros de leitura e realizará a pesquisa de homologia utilizando cada uma das seis novas seqüências de proteínas criadas e todas as novas seqüências de proteína do banco de dados criadas. Os HSPs nos resultados conterão alinhamentos aminoácido-aminoácido, ao invés de alinhamentos nucleotídeo-nucleotídeo. Os parâmetros fornecidos serão passados para o daemon e depois para o blastball através da linha de comandos. O wrapper retornará apenas aquelas colunas que forem especificamente solicitadas na consulta.

Related reference:

- “Origem de Dados de Documentum – Consultas de Exemplo” na página 188
- “Origem de Dados Excel – Consultas de Exemplo” na página 207

Dicas de Otimização para o Wrapper do BLAST

Executar o wrapper e o daemon no mesmo servidor pode eliminar possíveis gargalos na comunicação de rede.

Related reference:

- “Dicas e Considerações de Otimização para o Wrapper do Arquivo Estruturado de Tabela” na página 160

Mensagens para o Wrapper do BLAST

Esta seção lista e descreve mensagens que você pode encontrar quando trabalhar com o wrapper do BLAST.

Tabela 43. Mensagens emitidas pelo wrapper para o BLAST

| Código de Erro | Mensagem | Explicação |
|----------------|--|---|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. Razão "sqlno_crule_save_plans [100]:rc (-2144272209) Foi detectada uma lista de plano vazia".) | A consulta SQL enviada para o DB2 não foi processada pelo wrapper. Corrija a sintaxe e envie novamente. |
| SQL1816N | O wrapper "BLAST_WRAPPER" não pode ser utilizado para acessar o "tipo" de origem de dados (" <i><tipo do servidor></i> ") que você está tentando definir para o banco de dados federado. | A instrução CREATE SERVER utilizou um TYPE inválido. Ele deve ser um dos tipos suportados pelo BLAST. |
| SQL1817N | A instrução CREATE SERVER não identifica "versão" da origem de dados que foi definida para o banco de dados federado. | A instrução CREATE SERVER não especificou a versão. |
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Blast Wrapper". O texto e os tokens associados são "Não foi possível conectar ao daemon". | O wrapper do blast não pôde conectar ao daemon. O daemon pode não estar sendo executado. Ele pode estar configurado incorretamente. A máquina onde ele está sendo executado pode estar inacessível. |
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Blast Wrapper". O texto e os tokens associados são "Tempo limite do daemon do Blast expirou". | Nenhum resultado foi recebido do daemon antes do tempo limite especificado na instrução CREATE NICKNAME ser atingido. Aumente o tempo limite ou verifique se há algum problema com o daemon. |
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Blast Wrapper". O texto e os tokens associados são "Falha no daemon do Blast". | O daemon parou de se comunicar ou os resultados retornados não foram formatados apropriadamente. |

Tabela 43. Mensagens emitidas pelo wrapper para o BLAST (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Blast Wrapper". O texto e os tokens associados são "Erro desconhecido do daemon do blast". | O wrapper do blast não consegue reconhecer um código de erro que recebeu do daemon. A versão do daemon pode não ser compatível com a versão do wrapper. |
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Blast Wrapper". O texto e os tokens associados são "Renomeação de coluna não permitida". | Uma instrução ALTER NICKNAME foi emitida tentando renomear uma das colunas. Não é permitido renomear uma coluna. |
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Blast Wrapper". O texto e os tokens associados são "Erro de analisador XML". | O analisador Xerces está em um estado inválido ou lançou uma exceção. |
| SQL1823N | Não existe mapeamento de tipo de dados para o tipo de dados "<nome do tipo de dados>" do servidor "<nome do servidor>". | O tipo de dados especificado não é suportado por esta coluna. |
| SQL1881N | "DEFAULT" não é uma opção "COLUMN" válida para "<nome-da-coluna>" | A opção DEFAULT foi utilizada em uma coluna que não a suporta. Colunas apenas de saída e colunas de linha de definição não possuem valores padrão. |
| SQL1882N | A opção "DEFAULT" para "COLUMN" não pode ser definida para "<valor-da-opção>" para "<nome-da-coluna>". | O valor especificado para a opção DEFAULT é de um tipo incompatível para a coluna ou está formatado incorretamente. |

Related reference:

- "Mensagens para o Wrapper do Arquivo Estruturado de Tabela" na página 160
- "Mensagens para o Wrapper do Documentum" na página 194
- "Mensagens para o Wrapper do Excel" na página 211
- "Mensagens para o Wrapper do XML" na página 268

Capítulo 16. Configurando o Acesso às Origens de Dados do XML

Este capítulo explica o que é o XML, como incluir origens de dados do XML em seu sistema federado e lista as mensagens de erro associadas ao wrapper do XML.

O que É XML?

XML (Extensible Markup Language) é um formato universal para documentos e dados estruturados. Arquivos XML possuem uma extensão de arquivo `xml`. Como HTML, o XML utiliza marcações (palavras entre colchetes por `<` e `>`) para estruturar dados no documento. Uma amostra de documento XML é mostrada na Figura 7.

```
<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 7. Amostra de documento XML

Como Funciona o Wrapper XML

O wrapper XML permite a utilização de SQL para consultar os seguintes tipos de dados:

- Documentos XML externos que estão armazenados em um único arquivo
- Vários arquivos em um caminho do diretório
- Arquivos XML remotos que são referidos com um URI (Uniform Reference Identifier)
- Colunas relacionais

A Figura 8 mostra como o wrapper XML funciona com o sistema federado.

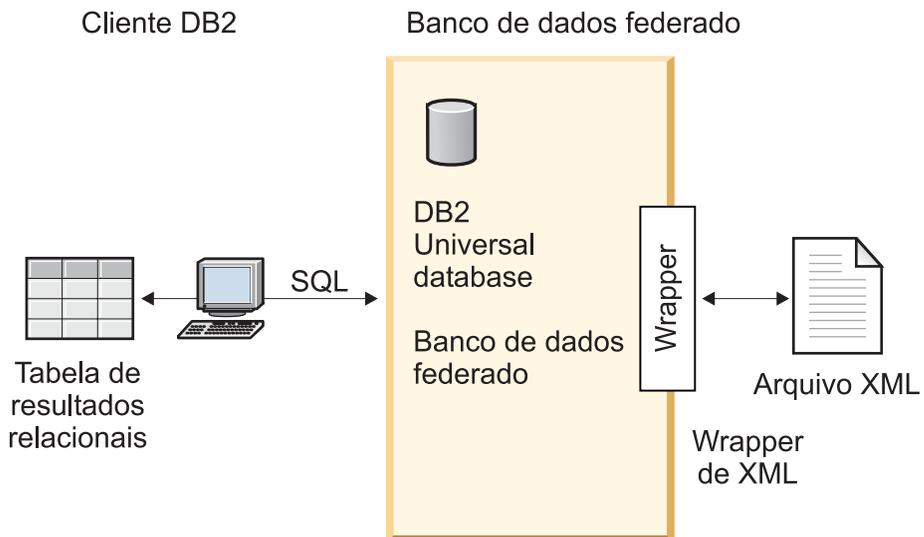


Figura 8. Como o wrapper do XML trabalha

Com o wrapper XML, você pode mapear dados XML de uma origem de dados externa para um esquema relacional que é composto de um conjunto de pseudônimos. A estrutura de um documento XML é logicamente equivalente a um esquema relacional no qual os elementos aninhados e repetidos são modelados como tabelas separadas com chaves externas.

Os pseudônimos correspondentes a um documento XML são organizados em uma estrutura de árvore na qual os pseudônimos filhos são mapeados para elementos que estão aninhados no elemento que corresponde ao pseudônimo pai.

Quando os elementos aninhados são repetidos ou possuem identidades distintas com estruturas complexas, você pode fornecer pseudônimos separados para cada elemento aninhado.

Os pseudônimos filho e pai são conectados por chaves primárias e externas que são geradas pelo wrapper.

Expressões XPath são utilizadas para mapear um documento XML para um esquema relacional composto por um conjunto de pseudônimos. XPath é um mecanismo de endereçamento para identificar as partes de um arquivo XML (por exemplo, os grupos de nós e atributos dentro de uma árvore de documento XML). A sintaxe XPath básica é semelhante a endereçamento de sistema de arquivos.

Cada pseudônimo é definido por uma expressão XPath que identifica os elementos XML que representam tuplas individuais e um conjunto de expressões XPath que especificam como extrair os valores de coluna de cada elemento.

Um Exemplo de Mapeamento de Documento XML:

O exemplo a seguir ilustra como o documento XML de amostra, mostrado na Figura 7 na página 247, é mapeado para um conjunto de pseudônimos, como os relacionamentos de pais e filhos são estabelecidos utilizando chaves primárias e externas, como expressões XPath são utilizadas para definir tuplas individuais e colunas em cada elemento do documento e como uma consulta pode ser executada no documento XML após o registro do documento no sistema federado.

O documento XML de amostra contém um conjunto de elementos do cliente. Cada elemento contém vários elementos de pedido e de pagamento.

Os elementos de pedido contém vários elementos de item.

O relacionamento entre os elementos são mostrados na Figura 9.

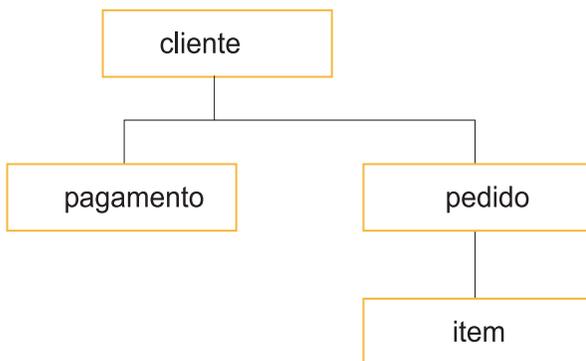


Figura 9. Estrutura de árvore da amostra do documento XML

Nesta estrutura, você pode utilizar a instrução `CREATE NICKNAME` para mapear o documento XML para um esquema relacional que inclui quatro pseudônimos:

- clientes
- pedidos
- pagamentos
- itens

Você define relacionamentos entre os pseudônimos, especificando cada pseudônimo como um pseudônimo pai ou um pseudônimo filho utilizando opções de colunas de chaves primária e externa. Cada pseudônimo pai deve ter uma coluna especial que é designada com uma opção de coluna de chave primária. Você define os filhos de um pseudônimo pai com a opção de coluna de chave externa que faz referência à coluna de chave primária de um pseudônimo pai. As colunas de pseudônimo primárias e externas designadas não correspondem aos dados em seu documento XML porque estas colunas de pseudônimos conterão as chaves que são geradas pelo wrapper. Um pseudônimo pode ter vários filhos, mas um pseudônimo pode ter apenas um pai. O pseudônimo raiz não possui pai.

Para o documento XML de amostra, o pseudônimo clientes possui uma chave primária definida e os pseudônimos pedidos, pagamentos e itens possuem chaves externas definidas que apontam para o pseudônimo pai. As chaves externas destes pseudônimos pedidos e pagamentos apontam para o pseudônimo clientes e a chave externa do pseudônimo itens aponta para o pseudônimo pedidos.

Para identificar os elementos XML que representam tuplas individuais, crie uma expressão XPath. Neste exemplo, todos os elementos do cliente são referidos utilizando a expressão XPath `//customer` e todos os elementos de pedido são referidos utilizando a expressão XPath `./order`. O ponto na expressão XPath `./order` indica que as tuplas de cada elemento de pedido estão aninhadas nas tuplas do elemento de cliente correspondente.

Você cria um conjunto de expressões XPath para especificar como extrair os valores de coluna de cada elemento. Neste exemplo, o atributo `id` dos elementos de cliente, agora uma coluna definida no pseudônimo, é referido utilizando a expressão XPath `./@id`. O elemento de nome dos elementos de cliente é referido utilizando a expressão XPath `./name` e o elemento de endereço dos elementos de cliente é referido utilizando a expressão XPath `./address/@street`.

Depois de mapear o documento XML para um conjunto de pseudônimos utilizando a instrução `CREATE NICKNAME`, defina cada pseudônimo como um pai ou filho, utilizando as chaves primária e externa, com expressões

XPath que definem tuplas individuais e colunas em cada elemento do documento. Você pode então executar consultas SQL no documento XML.

Related concepts:

- “O que São Arquivos com Estrutura de Tabela?” na página 151
- “O que É Documentum?” na página 165
- “O que É Excel?” na página 203
- “O que É BLAST?” na página 219
- “Associações de Dados entre Pseudônimos e Documentos XML” na página 254

Related tasks:

- “Incluindo XML a um Sistema Federado” na página 251

Incluindo XML a um Sistema Federado

Você pode utilizar uma origem de dados XML com o servidor federado, registrando um wrapper XML. Depois de registrar um wrapper XML, registre um servidor e pseudônimos correspondentes para permitir que o servidor federado recupere e processe dados XML.

Procedimento:

Para incluir uma origem de dados XML a um servidor federado:

1. Registre o wrapper utilizando a instrução CREATE WRAPPER. Registre o wrapper utilizando a instrução CREATE WRAPPER.
2. Opcional: Defina a variável de ambiente DB2_DJ_COMM para melhorar o desempenho da consulta.
3. Registre o servidor utilizando a instrução CREATE SERVER. Registre o servidor utilizando a instrução CREATE SERVER.
4. Registre pseudônimos utilizando a instrução CREATE NICKNAME. Registre pseudônimos utilizando a instrução CREATE NICKNAME.
5. Crie exibições federadas para pseudônimos não-raiz
Os pseudônimos raiz identificam os elementos no nível superior de um documento XML. Os pseudônimos não-raiz identificam os elementos nos níveis inferiores nesse elemento XML.

Você pode executar as instruções a partir do Centro de Controle do DB2 ou a partir de um processador da linha de comandos do DB2. Depois de adicionar o wrapper XML a seu sistema federado, você poderá executar consultas em uma origem de dados XML.

Related tasks:

- “Registrando o Wrapper do XML” na página 252
- “Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper XML” na página 253
- “Registrando o Servidor para uma Origem de Dados XML” na página 254
- “Registrando Pseudônimos para Origens de Dados XML” na página 260
- “Criando Exibições Federadas para Pseudônimos Não-raiz (Wrapper XML)” na página 266

Registrando o Wrapper do XML

O registro do wrapper XML faz parte da maior tarefa de adicionar XML a um sistema federado. Você deve registrar o wrapper para acessar uma origem de dados. Wrappers são mecanismos que os servidores federados utilizam para comunicar-se com e recuperar dados de origens de dados. Os wrappers são instalados no sistema como arquivos de biblioteca.

Você pode utilizar o wrapper XML nos seguintes sistemas operacionais:

- AIX
- HP-UX
- Linux
- Ambiente Operacional Solaris
- Windows NT

Procedimento:

Para registrar o wrapper do XML, envie a instrução CREATE WRAPPER.

Por exemplo, para registrar um wrapper XML no AIX chamado `my_xml`, a partir do arquivo de biblioteca padrão, `libdb2lxml.a`, emita a seguinte instrução:

```
CREATE WRAPPER my_xml LIBRARY 'libdb2lxml.a';
```

A próxima tarefa nesta seqüência de tarefas é definir a variável de ambiente `DB2_DJ_COMM` para o wrapper XML.

Related tasks:

- “Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper XML” na página 253
- “Depois de Instalar Wrappers Não Relacionais” no *DB2 Information Integrator Installation Guide*

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper XML

A definição da variável de perfil DB2_DJ_COMM do DB2 para o wrapper XML faz parte da maior tarefa de adicionar XML a um sistema federado. Para aprimorar o desempenho ao acessar documentos XML, opcionalmente, você pode definir a variável de perfil DB2_DJ_COMM do DB2. Esta variável determina se o servidor federado carrega o wrapper durante a inicialização.

Procedimento:

Para definir a variável de perfil DB2_DJ_COMM do DB2, envie o comando **db2set** com a biblioteca do wrapper que corresponde ao wrapper especificado na instrução CREATE WRAPPER associada.

Por exemplo:

```
db2set DB2_DJ_COMM='libdb21xml.a'
```

Verifique se não existem espaços em nenhum lado do sinal de igual (=).

O uso do processador do sistema aumenta quando o servidor federado carrega as bibliotecas do wrapper durante a inicialização do banco de dados. Para evitar o uso excessivo, especifique apenas as bibliotecas que deseja acessar.

A próxima tarefa nesta seqüência de tarefas é registrar o servidor para uma origem de dados XML.

Related concepts:

- “Environment Variables and the Profile Registry” no *Administration Guide: Implementation*

Related tasks:

- “Registrando o Servidor para uma Origem de Dados XML” na página 254

Related reference:

- “db2set - DB2 Profile Registry Command” em *Command Reference*

Registrando o Servidor para uma Origem de Dados XML

O registro do servidor para uma origem de dados XML faz parte da maior tarefa de adicionar XML a um sistema federado. Depois de registrar o wrapper, você deve registrar um servidor correspondente.

Restrições:

O wrapper do XML não utiliza as palavras-chave TYPE e VERSION. Ocorrerá um erro se estas palavras-chave forem utilizadas na instrução CREATE SERVER.

O wrapper XML não suporta sessões de passagem no sistema federado.

Procedimento:

Para registrar o servidor XML no sistema federado, emita a instrução CREATE SERVER.

Por exemplo:

```
CREATE SERVER xml_server WRAPPER my_xml;
```

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para origens de dados XML.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados XML” na página 260

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Associações de Dados entre Pseudônimos e Documentos XML

Pseudônimos correspondem à estrutura de árvore de dados do documento XML. Os pseudônimos pai e filho correspondem à estrutura raiz e a elementos aninhados da estrutura da árvore de dados. Estes pseudônimos pai e filho estão conectados por chaves primárias e externas que são especificadas com a instrução CREATE NICKNAME.

Cada pseudônimo é definido por expressões XPath que executam 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 no documento XML e determinam como estes valores correspondem às colunas de cada linha. O wrapper XML lê somente dados do documento XML. O wrapper XML não atualiza estes dados.

Ao criar um pseudônimo, você escolhe opções que especificam a associação entre o pseudônimo e o documento XML. Os pseudônimos são associados aos documentos XML de maneira fixa ou com nomes de origens especificados.

Com uma associação fixa, o pseudônimo representa dados de documentos XML específicos. Estes documentos XML incluem:

Um arquivo local

Especifique 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 do documento XML para o pseudônimo. Todos os arquivos XML devem ter a mesma configuração. Se qualquer arquivo XML no diretório tiver uma configuração que seja diferente da configuração do pseudônimo, o wrapper XML retornará valores nulos quando processar esse arquivo de dados XML. O diretório deve ser local para o servidor federado ou acessível a partir de um sistema de arquivos compartilhado.

Nota: Ao varrer o diretório, o wrapper XML mantém e analisa somente os arquivos com uma extensão .xml. O wrapper XML ignora todos os demais arquivos, incluindo arquivos com uma extensão .txt, arquivos com uma extensão .xsd e arquivos sem extensões.

Utilize a opção FILE_PATH da 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 a origem de dados é especificada durante a execução da consulta, você pode utilizar o pseudônimo para representar dados de qualquer origem de documento XML cujo esquema corresponde à definição do pseudônimo. Estes documentos XML incluem:

Uniform Reference Identifiers

Um arquivo XML remoto ao qual um URI se refere fornece dados do documento XML para o pseudônimo. (Especifique esta origem de documento utilizando a opção de coluna de pseudônimo DOCUMENT 'URI').

Colunas relacionais

Colunas de uma tabela, exibição ou pseudônimo relacional são utilizadas como entrada para seu documento XML. (Especifique esta origem de documento utilizando a opção de coluna de pseudônimo DOCUMENT 'COLUMN'.)

Arquivo

Um único arquivo que contém dados XML é fornecido como entrada durante a execução da consulta. (Especifique esta origem de documento na opção de coluna de pseudônimo DOCUMENT 'FILE').

Diretório

Vários arquivos XML em um caminho de diretório especificado fornecem os dados durante a execução da consulta. (Especifique esta origem de documento utilizando a opção de coluna de pseudônimo DOCUMENT 'DIRECTORY'.)

Especifique a opção da coluna DOCUMENT para indicar que a origem de dados é fornecida no momento da consulta. Especifique URI, COLUMN, FILE ou DIRECTORY com a coluna DOCUMENT para indicar o tipo de origem de documento XML.

Você não pode especificar uma opção FILE_PATH ou uma opção DIRECTORY_PATH com uma opção de coluna DOCUMENT.

Independentemente do tipo de dados que está sendo utilizado (dados em um formato fixo ou dados de nomes de origens que são especificados no momento da consulta), você pode especificar a opção STREAMING para que o wrapper XML separe dados de documento XML em fragmentos. O wrapper XML processa o fluxo resultante de dados XML e extrai as informações solicitadas por um fragmento de consulta. O wrapper XML analisa um fragmento de cada vez. Como os fragmentos são analisados um de cada vez, a utilização de memória total é reduzida, mas o tempo de processamento requerido para executar toda a consulta aumenta, dependendo da capacidade da memória do servidor. Portanto, utilize a opção STREAMING para analisar somente grandes documentos XML (documentos de 50 megabytes ou mais).

Você também pode escolher valores de opção de pseudônimo que ajudam a otimizar consultas que recuperam grandes quantidades de dados XML ou de dados que contêm vários elementos aninhados. Estas opções incluem:

- INSTANCE_PARSE_TIME
- XPATH_EVAL_TIME
- NEXT_TIME

Você pode definir 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 para analisar os dados nas linhas do documento XML.

Related concepts:

- “O que É XML?” na página 247
- “O Recurso de Modelo de Custo para o Wrapper XML” na página 257
- “Dicas de Otimização para o Recurso de Modelo de Custo XML” na página 258

Related tasks:

- “Registrando Pseudônimos para Origens de Dados XML” na página 260

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper XML” na página 375
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper XML” na página 261

O Recurso de Modelo de Custo para o Wrapper XML

O wrapper XML fornece um recurso de modelo de custo para otimizar consultas em pseudônimos que correspondem a seus documentos de origem XML.

Ao criar um pseudônimo utilizando a instrução CREATE NICKNAME, você pode especificar os seguintes parâmetros como valores de opções de pseudônimos para suportar o recurso de modelo de custo:

- INSTANCE_PARSE_TIME
- XPATH_EVAL_TIME

Você pode utilizar os valores padrão para estes parâmetros. Ou pode definir os valores para estes parâmetros para otimizar consultas nos pseudônimo raiz e não-raiz que forem criados.

O parâmetro INSTANCE_PARSE_TIME é o período de tempo (em milissegundos) requerido para leitura e análise de um elemento raiz de produção de linhas do pseudônimo raiz (por exemplo, customers), incluindo todos os elementos não-raiz de produção de linha contidos (por exemplo, todos os elementos que correspondem a pedidos, pagamentos e itens de cada cliente). O wrapper XML constrói uma estrutura na memória para representar estes elementos raiz e não-raiz de produção de linhas.

O parâmetro `XPATH_EVAL_TIME` é o período de tempo (em milissegundos) requerido para avaliar as expressões XPath que localizam dados correspondentes a uma linha do pseudônimo. As expressões XPath avaliadas incluem as expressões XPath que localizam as linhas reais e as expressões XPath que localizam valores de colunas nessas linhas.

Related concepts:

- “O que É XML?” na página 247
- “Associações de Dados entre Pseudônimos e Documentos XML” na página 254
- “Dicas de Otimização para o Recurso de Modelo de Custo XML” na página 258

Related reference:

- “Sintaxe da Instrução `CREATE NICKNAME - Wrapper XML`” na página 375
- “Instrução `CREATE NICKNAME - Exemplos para o Wrapper XML`” na página 261

Dicas de Otimização para o Recurso de Modelo de Custo XML

O recurso de modelo de custo para o wrapper XML ajuda a otimizar consultas nos pseudônimos criados.

O recurso de modelo de custo utiliza os seguintes parâmetros da instrução `CREATE NICKNAME`:

- `INSTANCE_PARSE_TIME`
- `XPATH_EVAL_TIME`

Você pode especificar valores para estes parâmetros ao emitir uma instrução `CREATE NICKNAME` para registrar um pseudônimo para uma origem de dados XML.

O recurso de modelo de custo utiliza estes valores de parâmetros ao determinar o período de tempo requerido para analisar dados em cada linha de um documento de origem XML e para avaliar a expressão XPath do pseudônimo.

Você pode utilizar os valores padrão para estes parâmetros. No entanto, se desejar otimizar consultas em estruturas de origem XML grandes ou complexas para os pseudônimos criados, utilize o seguinte exemplo como guia.

Um Exemplo de Otimização de uma Grande Consulta:

Suponha que seu documento XML tenha um esquema relacional com quatro pseudônimos:

- customers
- orders
- payments
- items

Suponha também que o pseudônimo customers seja o pseudônimo raiz.

Execute consultas em cada pseudônimo. Execute cada consulta em uma amostra dos dados XML que seja típica de seu ambiente.

Por exemplo:

```
SELECT * from customers;  
SELECT * from orders;  
SELECT * from payments;  
SELECT * from items;
```

Anote o período de tempo (em milissegundos) requerido 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 requerido para executar consultas.) Anote também o número de tuplas retornadas.

Para cada pseudônimo, utilize as seguintes fórmulas para determinar os valores corretos para os parâmetros `INSTANCE_PARSE_TIME` e `XPATH_EVAL_TIME`:

```
INSTANCE_PARSE_TIME = (75% X run time of SELECT * query) ÷ number of tuples returned  
XPATH_EVAL_TIME     = (25% X run time of SELECT * query) ÷ number of tuples returned
```

Para o pseudônimo raiz (no exemplo, customers), utilize os valores calculados para os parâmetros `INSTANCE_PARSE_TIME` e `XPATH_EVAL_TIME`.

Para pseudônimos não-raiz (no exemplo, orders, payments e items), utilize o valor calculado para o parâmetro `XPATH_EVAL_TIME`. O valor de parâmetro `INSTANCE_PARSE_TIME` não é aplicável para pseudônimos não-raiz.

Você pode utilizar estas fórmulas como um guia para ajustar consultas. Os valores ideais para estes parâmetros também dependem da complexidade dos documentos de origem XML e da velocidade do processador que está sendo utilizado.

Related concepts:

- “O que É XML?” na página 247
- “Associações de Dados entre Pseudônimos e Documentos XML” na página 254
- “O Recurso de Modelo de Custo para o Wrapper XML” na página 257

Related reference:

- “db2batch - Benchmark Tool Command” no *Command Reference*

Registrando Pseudônimos para Origens de Dados XML

O registro de pseudônimos para origens de dados XML faz parte da maior tarefa de adicionar um XML a um sistema federado. Você deve criar pseudônimos que correspondem à estrutura de árvore de sua origem de dados XML. Os pseudônimos pai correspondem à estrutura raiz da árvore. Os pseudônimos filhos correspondem aos elementos aninhados no elemento para o pseudônimo pai.

Pré-requisito:

A página de código do banco de dados deve corresponder ao conjunto de caracteres dos arquivos de origem XML.

Restrição:

Não são suportados espaços de nomes.

Procedimento:

Para registrar pseudônimos para origens de dados XML, emita uma instrução CREATE NICKNAME.

A próxima tarefa nesta seqüência de tarefas é criar exibições federadas para pseudônimos não-raiz (wrapper XML).

Related tasks:

- “Criando Exibições Federadas para Pseudônimos Não-raiz (Wrapper XML)” na página 266

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper XML” na página 375
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper XML” na página 261

Instrução CREATE NICKNAME - Exemplos para o Wrapper XML

Este tópico fornece vários exemplos que mostram como utilizar a instrução CREATE NICKNAME para registrar pseudônimos para o wrapper XML. Este tópico inclui um exemplo completo, que mostra como criar pseudônimos pai e filho, exemplos para opções de colunas específicas e exemplos que mostram a utilização de exibições.

Exemplo completo:

O exemplo a seguir mostra como criar pseudônimos para origens de dados XML utilizando o arquivo XML de amostra mostrado na Figura 10.

```
<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 10. Amostra de arquivo XML

Para criar o pseudônimo pai, customers, 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)  OPTIONS(PRIMARY_KEY 'YES'))
```

```

FOR SERVER xml_server
OPTIONS(DIRECTORY_PATH '/home/db2user',
        XPATH '//customer', STREAMING 'YES');

```

Esta instrução cria o pseudônimo customers sobre vários arquivos XML no caminho do diretório especificado, /home/db2user. A opção STREAMING indica que a origem de dados XML é separada e processada pelo nó (neste exemplo, pelo registro do cliente).

Agora você pode criar pseudônimos para os filhos do pseudônimo customers (orders, payments e items).

Emita a seguinte instrução de pseudônimo para criar o pseudônimo 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)  OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
FOR SERVER xml_server
OPTIONS( XPATH './order');

```

Emita a seguinte instrução de pseudônimo para criar o pseudônimo payments.

```

CREATE NICKNAME payments
(
  number  INTEGER      OPTIONS(XPATH './number'),
  date    VARCHAR(10)  OPTIONS(XPATH './date'),
  cid     VARCHAR(16)  OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
FOR SERVER xml_server
OPTIONS( XPATH './payment');

```

Emita a seguinte instrução de pseudônimo para criar o pseudônimo items.

```

CREATE NICKNAME items
(
  name     VARCHAR(20)  OPTIONS(XPATH './name'),
  quantity INTEGER      OPTIONS(XPATH './@quant'),
  oid     VARCHAR(16)  OPTIONS(FOREIGN_KEY 'ORDERS'))
FOR SERVER xml_server
OPTIONS( XPATH './item');

```

Exemplos de opções de colunas:

Os exemplos de opções de colunas mostram como criar pseudônimos utilizando as opções de colunas DOCUMENT.

O exemplo CREATE NICKNAME a seguir mostra a utilização da opção de coluna 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)   OPTIONS(PRIMARY_KEY 'YES')
  FOR SERVER xml_server
  OPTIONS(XPATH '/customer');

```

Você pode executar a seguinte consulta no pseudônimo customers, especificando a localização do documento XML na cláusula WHERE:

```

SELECT * FROM customers WHERE doc='/home/db2user/Customers.xml';

```

O exemplo CREATE NICKNAME a seguir mostra a utilização da opção de coluna 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)   OPTIONS(PRIMARY_KEY 'YES')
  FOR SERVER xml_server
  OPTIONS(XPATH '/customer');

```

Você pode executar a seguinte consulta no pseudônimo customers:

```

SELECT name FROM customers WHERE doc = '/home/data/xml';

```

Esta consulta recupera os documentos XML que estão localizados no caminho do diretório /home/data/xml, que está especificado na cláusula WHERE.

O exemplo CREATE NICKNAME a seguir mostra a utilização da opção de coluna de pseudônimo 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)   OPTIONS(PRIMARY_KEY 'YES')
  FOR SERVER xml_server
  OPTIONS(XPATH '/customer');

```

Você pode executar a seguinte consulta no pseudônimo customers para recuperar os dados XML da localização remota:

```

SELECT * FROM customers WHERE doc = 'http://www.lg-mv.org/foo.xml';

```

O exemplo CREATE NICKNAME a seguir mostra a utilização da opção de coluna de pseudônimo 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 '//employee');

```

Você pode executar uma das seguintes consultas no pseudônimo 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 populada com os dados XML.

Exemplos de exibição:

Os exemplos de exibição mostram como criar exibições para pseudônimos não-raiz para descrever documentos de origem XML. Nestes exemplos, suponha que os pseudônimos do arquivo de amostra mostrados na Figura 11 na página 265 tenham sido criado anteriormente como customers, orders, payments e items.

```

<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 11. Amostra de arquivo XML

O exemplo a seguir mostra como criar uma exibição para a ordem de pseudônimos não-raiz:

```

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 exibição para o pagamento de pseudônimo não-raiz:

```

CREATE VIEW payment_view AS
SELECT p.amount, p.date, c.cid
FROM customers c, payments p
WHERE c.cid = p.cid;

```

O exemplo a seguir mostra como criar uma exibição para o item de pseudônimo não-raiz:

```

CREATE VIEW item_view AS
SELECT it.quantity, it.name, o.oid
FROM customers c, orders o, items i
WHERE c.cid = o.cid AND o.oid = i.oid;

```

Consultas enviadas para estas exibições são processadas corretamente porque a união do caminho para o diretório raiz está presente.

Por exemplo, a consulta a seguir une as quantidades de pedidos e pagamentos de clientes a partir da mesma data:

```
SELECT o.amount, p.amount
FROM order_view o, payment_view p
WHERE p.date = o.date AND
      p.cid = o.cid;
```

Related tasks:

- “Registrando Pseudônimos para Origens de Dados XML” na página 260

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper XML” na página 375

Criando Exibições Federadas para Pseudônimos Não-raiz (Wrapper XML)

A criação de exibições federadas para pseudônimos não-raiz (wrapper XML) faz parte da maior tarefa de adicionar XML a um sistema federado.

Você pode definir exibições federadas sobre a hierarquia de pseudônimos que descrevem um documento XML. A definição de exibições federadas assegura que as consultas que unem partes de uma hierarquia de pseudônimos XML (não incluindo o pseudônimo raiz e consultas que unem colunas diferentes das colunas especiais PRIMARY_KEY e FOREIGN_KEY) sejam executadas corretamente.

Procedimento:

Para definir exibições federadas que incluem todos os predicados requeridos e um caminho completo para o diretório raiz, siga estas etapas:

1. Defina uma exibição para cada pseudônimo não-raiz como uma união de todos os pseudônimos no caminho para a raiz.
2. Na cláusula WHERE, faça a união dos predicados sobre as colunas PRIMARY_KEY e FOREIGN_KEY.
3. Na lista SELECT, inclua todas as colunas de pseudônimos não-raiz, exceto a coluna que foi designada com a opção de coluna de pseudônimos FOREIGN_KEY.
4. Na lista SELECT, inclua a coluna do pseudônimo pai designada com a opção PRIMARY_KEY.

Related reference:

- “Instrução CREATE NICKNAME - Exemplos para o Wrapper XML” na página 261

Origem de Dados XML - Consultas de Exemplo

Este tópico fornece várias consultas de amostra que utilizam os pseudônimos customers, orders e items. Estes pseudônimos foram registrados anteriormente utilizando instruções CREATE NICKNAME.

A consulta a seguir exibe todos os nomes dos clientes:

```
SELECT name FROM customers;
```

A consulta a seguir exibe todos os registros nos quais o nome do cliente é Chang:

```
SELECT * FROM customers where name='Chang';
```

A consulta a seguir exibe os nomes de clientes e valores para cada pedido de cada cliente:

```
SELECT c.name, o.amount FROM customers c, orders o where c.cid=o.cid;
```

Você deve especificar a união, c.cid=o.cid para indicar o relacionamento entre pai/filho entre o pseudônimo customers e o pseudônimo orders.

A consulta a seguir seleciona os endereços do cliente, valores de pedidos e nomes de itens para cada pedido e item de cada cliente:

```
SELECT c.address, o.amount, i.name FROM customers c, orders o, items i  
WHERE c.cid=o.cid AND o.oid=i.oid;
```

Você deve especificar as duas uniões para manter os relacionamentos entre pai/filho.

Os exemplos a seguir mostram como gravar consultas utilizando um pseudônimo que especifica uma opção de coluna DOCUMENT em vez de uma opção de pseudônimo FILE_PATH. A instrução CREATE NICKNAME correspondente que é utilizada para criar o pseudônimo customers é mostrada a seguir:

```
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 '/customer');
```

A consulta a seguir seleciona todos os dados a partir do arquivo XML Customers.xml com um caminho de arquivo de /home/db2user/Customers.xml:

```
SELECT * FROM customers WHERE doc='/home/db2user/Customers.xml';
```

A consulta a seguir seleciona nomes de clientes e datas de seus pedidos a partir do arquivo Customers.xml para cada pedido com um valor acima de 1000:

```
SELECT c.name, o.date FROM customers c, orders o  
WHERE c.doc='/home/db2user/Customers.xml' AND o.amount > 1000;
```

O caminho do arquivo de /home/db2user/Customers.xml especifica a localização do arquivo Customers.xml.

Related reference:

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper XML” na página 375
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper XML” na página 261

Mensagens para o Wrapper do XML

Este tópico descreve mensagens que podem ser encontradas ao trabalhar com o wrapper para XML. Para obter mais informações sobre mensagens, consulte o *DB2 Message Reference*.

Tabela 44. Mensagens emitidas pelo wrapper para XML

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL0405N | O literal numérico "<nome_da_coluna>" não é válido porque seu valor está fora do intervalo. | O literal numérico especificado não está no intervalo aceitável. Verifique o tipo de dados da coluna na instrução CREATE NICKNAME. |
| SQL0408N | Um valor não é compatível com o tipo de dados do destino da atribuição. O nome de destino é "<nome_da_coluna>." | O tipo de dados do valor que está sendo atribuído à coluna não é compatível com o tipo de dados declarado do destino de atribuição. Verifique o tipo de dados da coluna na instrução CREATE NICKNAME. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Erro ao criar o objeto do wrapper.") | Ocorreu um erro ao criar um novo objeto do wrapper. Entre em contato com o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "<xerces_xalan_error_message>.") | Ocorreu um erro durante uma chamada para uma função Xerces ou Xalan. Verifique o documento XML. Se o documento for bem estruturado, consulte a documentação do Xalan para obter mais informações sobre a mensagem de erro. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "XalanDOMException: exception code is <exception_code>.") | Uma exceção XalanDOMException ocorreu. Consulte a documentação do Xalan para obter mais informações sobre o código da exceção. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "XMLException: <exception_error_message>.") | Ocorreu uma exceção XMLException. Consulte a documentação do Xalan para obter mais informações sobre o código da exceção. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "XSLEException: <exception_error_message>.") | Ocorreu uma exceção XSLEException. Consulte a documentação do Xalan para obter mais informações sobre o código da exceção. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "SAXParseException: <exception_error_message>.") | Ocorreu uma exceção SAXParseException. Consulte a documentação do Xalan para obter mais informações sobre o código da exceção. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Erro ao obter valor do nó.") | Xalan tentou acessar um nó inválido. Entre em contato com o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Erro ao analisar documento XML.") | Ocorreu um erro ao analisar o documento XML. Verifique o documento XML. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Erro ao obter elemento raiz de documento XML.") | Depois de analisar o documento XML, Xalan tentou recuperar o elemento raiz mas falhou. Verifique o documento XML. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Especificada exceção ao avaliar expressão XPath.") | Xalan gerou uma exceção não especificada ao avaliar uma expressão XPath. Verifique o documento XML e consulte a documentação do Xalan. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Exceção não especificada ao obter valor do nó.") | Xalan gerou uma exceção não especificada ao recuperar um valor do nó. Verifique o documento XML e consulte a documentação do Xalan. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Exceção não especificada ao analisar documento de entrada.") | Xalan gerou uma exceção não especificada ao analisar o documento XML. Verifique o documento XML e consulte a documentação do Xalan. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|--|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Erro ao avaliar cardinalidade.") | Entre em contato com o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "<SOAP_error_message>.") | A biblioteca SOAP emitiu um erro. Se não puder resolver o erro da instrução SQL, contacte o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "URI Inválida.") | O wrapper não pode acessar a URL especificada. Verifique se a URL está acessível. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Conteúdo de documento XML inválido.") | O conteúdo do documento XML não é válido. Verifique se o documento está bem estruturado. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Envelope SOAP inválido.") | O envelope SOAP não é válido. Verifique a sintaxe e o conteúdo. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Erro de alocação de memória.") | Ocorreu um erro durante a alocação de memória. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e token associados são "Formato DATE incorreto." | Um valor de data no documento XML não possui o formato correto. O formato válido para valores de data é aaaa-mm-dd. Verifique o documento XML. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|--|
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Tipo de dados de coluna não suportado." | Uma coluna de pseudônimo possui um tipo de dados não suportado. Verifique a instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Cláusula TYPE não suportada." | A instrução CREATE SERVER contém uma cláusula TYPE. Esta cláusula não é suportada pelo wrapper do XML. Remova a cláusula. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Cláusula VERSION não suportada." | A instrução CREATE SERVER contém uma cláusula VERSION. Esta cláusula não é suportada pelo wrapper do XML. Remova a cláusula. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Utilização inválida de predicado com a coluna DOCUMENT." | A consulta contém um predicado com operandos incorretos. Verifique os predicados na consulta. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Utilização inválida de predicado com a coluna FOREIGN_KEY." | A consulta contém um predicado com operandos incorretos. Verifique os predicados na consulta. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Utilização inválida de predicado com a coluna PRIMARY_KEY." | A consulta contém um predicado com operandos incorretos. Verifique os predicados na consulta. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções XPATH e DOCUMENT não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções XPATH e FOREIGN_KEY não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções XPATH e PRIMARY_KEY não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções DOCUMENT e FOREIGN_KEY não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções DOCUMENT e PRIMARY_KEY não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções FOREIGN_KEY e PRIMARY_KEY não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opção de coluna ausente." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opção de coluna DOCUMENT não exclusiva." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opção de coluna FOREIGN_KEY não exclusiva." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opção de coluna PRIMARY_KEY não exclusiva." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique a sintaxe da instrução. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Valor de opção DOCUMENT inválido." | O valor da opção DOCUMENT especificado na instrução CREATE NICKNAME é inválido. O valor deve ser FILE. Verifique a instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Valor de opção PRIMARY_KEY inválido." | O valor da opção PRIMARY_KEY especificado na instrução CREATE NICKNAME é inválido. O valor deve ser YES. Verifique a instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Valor de opção FOREIGN_KEY inválido." | O valor da opção FOREIGN_KEY especificado na instrução CREATE NICKNAME é inválido. O valor não corresponde a nenhum pseudônimo pai. Verifique a instrução CREATE NICKNAME. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|--|
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções FILE_PATH e DOCUMENT não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. As opções FILE_PATH e DOCUMENT não podem ser especificadas ao mesmo tempo. Verifique a sintaxe da instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções FILE_PATH e SOAP não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. As opções FILE_PATH e SOAP não podem ser especificadas ao mesmo tempo. Verifique a sintaxe da instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções DIRECTORY_PATH e SOAP não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. As opções DIRECTORY_PATH e SOAP não podem ser especificadas ao mesmo tempo. Verifique a sintaxe da instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções FILE_PATH e DIRECTORY_PATH não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. As opções FILE_PATH e DIRECTORY_PATH não podem ser especificadas ao mesmo tempo. Verifique a sintaxe da instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções VALIDATE e STREAMING não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. As opções VALIDATE e STREAMING não podem ser especificadas ao mesmo tempo. Verifique a sintaxe da instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções FILE_PATH e FOREIGN_KEY não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. As opções FILE_PATH e FOREIGN_KEY não podem ser especificadas ao mesmo tempo. Verifique a sintaxe da instrução CREATE NICKNAME. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|--|
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opções DIRECTORY_PATH e FOREIGN_KEY não compatíveis." | A instrução CREATE NICKNAME não está correta conforme especificado. As opções DIRECTORY_PATH e FOREIGN_KEY não podem ser especificadas ao mesmo tempo. Verifique a sintaxe da instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Valor de opção XPATH inválido com STREAMING ativado." | A expressão XPATH do pseudônimo não será válida quando você ativar o recurso STREAMING. Verifique na opção XPATH valores inválidos, como /, ./ e //. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Impossível ler arquivo XML." | O caminho do arquivo especificado na instrução CREATE NICKNAME ou na consulta é inválido. O arquivo especificado não existe. Verifique a instrução CREATE NICKNAME e a consulta. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Impossível abrir o diretório." | O caminho do diretório especificado na instrução CREATE NICKNAME ou na consulta é inválido. O diretório especificado não existe. Verifique a instrução CREATE NICKNAME e a consulta. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Referência a dados XML ausente." | A instrução CREATE NICKNAME deve conter uma referência aos dados XML. Verifique a instrução CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreo>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opção de coluna DOCUMENT com valor 'SOAP' ausente." | A instrução CREATE NICKNAME não está correta conforme especificado. Verifique o valor da opção DOCUMENT. O valor deve ser SOAP. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|--|
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opção SOAP ausente." | A instrução CREATE NICKNAME não está correta conforme especificado. Você deve especificar a opção SOAP. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "INSTANCE_PARSE_TIME apenas para pseudônimos raiz." | A instrução CREATE NICKNAME não está correta conforme especificado. Você pode especificar um valor INSTANCE_PARSE_TIME apenas para pseudônimos raiz. Verifique a sintaxe de CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opção VALIDATE apenas para pseudônimos raiz." | A instrução CREATE NICKNAME não está correta conforme especificado. Você pode definir a opção VALIDATE como YES apenas se o pseudônimo especificado for um pseudônimo raiz. Verifique a sintaxe de CREATE NICKNAME. |
| SQL1822N | Código de erro inesperado "<ponto_do_rastreio>" recebido da origem de dados "Wrapper do XML". Texto e tokens associados são "Opção STEAMING apenas para pseudônimos raiz." | A instrução CREATE NICKNAME não está correta conforme especificado. Você pode definir a opção STREAMING como YES apenas se o pseudônimo especificado for um pseudônimo raiz. Verifique a sintaxe de CREATE NICKNAME. |
| SQL1823N | Não existe nenhum mapeamento de tipo de dados para o tipo de dados "<data_type_name>" a partir do servidor "<server_name>." | A instrução CREATE NICKNAME não está correta conforme especificado. Um tipo de dados da coluna é inválido. Verifique a sintaxe de CREATE NICKNAME. |
| SQL1881N | "<option_name>" não é uma opção "<option_type>" válida para "<object_name>." | A opção especificada pode não existir ou pode não ser válida para esta origem de dados. Verifique a instrução CREATE NICKNAME. |
| SQL1881N | "DIRECTORY_PATH" não é uma opção "NICKNAME" válida para "<object_name>." | O valor da opção DIRECTORY_PATH especificado na instrução CREATE NICKNAME é inválido. O diretório especificado deve ser um diretório raiz. Verifique a instrução CREATE NICKNAME. |

Tabela 44. Mensagens emitidas pelo wrapper para XML (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|--|
| SQL1882N | A opção "VALIDATE" de "nickname" não pode ser definida como "<option_value>" para "<object_name>." | O valor da opção VALIDATE especificado na instrução CREATE NICKNAME é inválido. Este valor deve ser YES ou NO. Verifique a instrução CREATE NICKNAME. |
| SQL1882N | A opção "STREAMING" de "nickname" não pode ser definida como "<option_value>" para "<object_name>." | O valor da opção STREAMING especificado na instrução CREATE NICKNAME é inválido. Este valor deve ser YES ou NO. Verifique a instrução CREATE NICKNAME. |
| SQL1883N | "<option_name>" é uma opção "<option_type>" requerida para "<object_name>." | Uma opção do DB2 requerida não foi especificada. Verifique a instrução CREATE NICKNAME. |

Related reference:

- "Mensagens para o Wrapper do Arquivo Estruturado de Tabela" na página 160
- "Mensagens para o Wrapper do Documentum" na página 194
- "Mensagens para o Wrapper do Excel" na página 211
- "Mensagens para o Wrapper do BLAST" na página 243
- "Síntaxe da Instrução CREATE NICKNAME - Wrapper XML" na página 375

Capítulo 17. Configurando o Acesso às Origens de Dados do Entrez

Este capítulo explica o que é o Entrez, como incluir origens de dados do Entrez em seu sistema federado e lista as mensagens de erro associadas ao wrapper do Entrez.

O que É Entrez?

Entrez é um sistema de consulta e de recuperação desenvolvido pelo NCBI (National Center for Biotechnology Information). Você pode utilizar o Entrez para acessar vários bancos de dados vinculados hospedados pelo NCBI.

Estes bancos de dados incluem:

- PubMed (literatura biomédica)
- Nucleotide (um banco de dados de seqüência também chamado GenBank)
- OMIM (Online Mendelian Inheritance in Man from John Hopkins University)
- Genome (montagens completas de genomas)

Você pode acessar todos os bancos de dados Entrez através de um conjunto uniforme de ferramentas baseadas na Web. O wrapper Entrez utiliza estas ferramentas para federar os bancos de dados Entrez no ambiente do DB2[®]. Embora a interface do Entrez suporte muitos bancos de dados, o wrapper Entrez suporta apenas PubMed e Nucleotide.

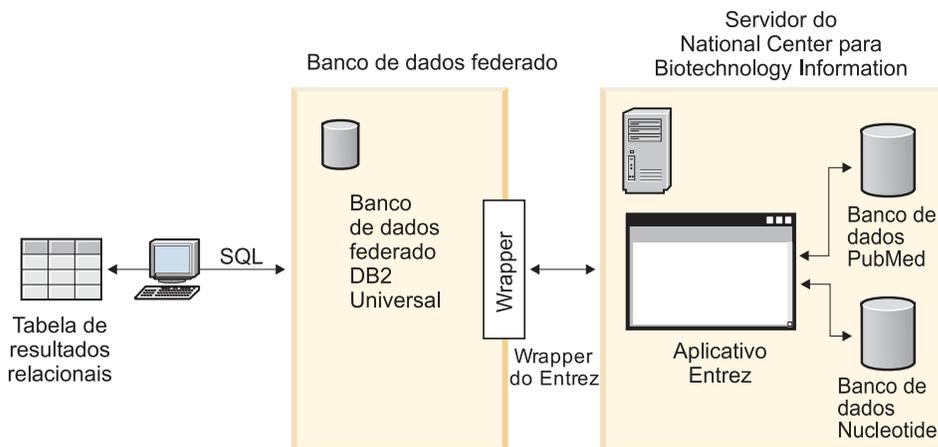


Figura 12. Como funciona o wrapper Entrez

Muitos elementos do wrapper Entrez são comuns a todos os bancos de dados. Estes elementos incluem:

- Conectividade com o NCBI através da Web e dos utilitários Entrez ESearch e EFetch
- Mapeamento de dados XML hierárquicos em tabelas relacionais
- Uniões entre tabelas relacionadas através da tecnologia do wrapper XML

Related concepts:

- “O que São Arquivos com Estrutura de Tabela?” na página 151
- “O que É Documentum?” na página 165
- “O que É Excel?” na página 203
- “O que É BLAST?” na página 219
- “O que É XML?” na página 247

Related tasks:

- “Adicionando Entrez a um Sistema Federado” na página 280

Adicionando Entrez a um Sistema Federado

Você pode executar instruções a partir do Processador da Linha de Comandos do DB2. Depois de adicionar o wrapper Entrez a seu sistema federado, você pode executar consultas em uma origem de dados Entrez.

Procedimento:

Para adicionar a origem de dados Entrez a um servidor federado:

1. Registre funções personalizadas, emitindo uma instrução CREATE FUNCTION.
2. Registre o wrapper, emitindo uma instrução CREATE WRAPPER.
3. Opcional: Defina a variável de ambiente DB2_DJ_COMM para aprimorar o desempenho das consultas.
4. Registre o servidor, emitindo uma instrução CREATE SERVER.
5. Registre pseudônimos, emitindo uma instrução CREATE NICKNAME.

Related tasks:

- “Registrando Funções Personalizadas para o Wrapper Entrez” na página 281

Registrando Funções Personalizadas para o Wrapper Entrez

O registro de funções personalizadas para o wrapper Entrez faz parte da maior tarefa de adicionar Entrez a um sistema federado. Após o registro de funções personalizadas, é necessário registrar o wrapper.

Restrições:

- Todas as funções personalizadas para o wrapper Entrez devem ser registradas com o Entrez de nome do esquema.
- É necessário registrar cada função personalizada uma vez para cada banco de dados DB2 que tenha o wrapper Entrez instalado.

Procedimento:

Para registrar funções personalizadas, emita a instrução CREATE FUNCTION com a palavra-chave AS TEMPLATE.

O nome completo de cada função é Entrez.<function-name>.

O exemplo a seguir registra uma versão da função CONTAINS:

```
CREATE FUNCTION entrez.contains (varchar(), varchar())  
  RETURNS INTEGER AS TEMPLATE;
```

Para ajudá-lo a registrar funções personalizadas, o arquivo de amostra, create_function_mappings.ddl, é fornecido no diretório samples/lifesci/entrez. Esse arquivo contém definições para cada função personalizada. Você pode executar esse arquivo DDL para registrar as funções personalizadas de cada banco de dados DB2 com o wrapper Entrez instalado.

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper Entrez.

Related reference:

- “CREATE FUNCTION (Sourced or Template) statement” no *SQL Reference, Volume 2*
- “Funções Personalizadas e Consultas Entrez” na página 286
- “Tabela de Funções Personalizadas - Wrapper Entrez” na página 291

Registrando o Wrapper Entrez

O registro do wrapper Entrez faz parte da maior tarefa de adicionar Entrez a um sistema federado. Você deve registrar o wrapper para acessar uma origem de dados. Wrappers são mecanismos que os servidores federados utilizam para se comunicar com as origens de dados e recuperar dados delas. Instale wrappers em seu sistema como arquivos de biblioteca.

Procedimento:

Para registrar o wrapper Entrez, emita uma instrução CREATE WRAPPER.

Por exemplo, para criar um wrapper Entrez no AIX chamado `entrez_wrapper`, a partir do arquivo de biblioteca padrão, `libdb2lsentrez.a`, envie a seguinte instrução:

```
CREATE WRAPPER entrez_wrapper LIBRARY 'libdb2lsentrez.a'  
  OPTIONS(EMAIL 'jeff@someplace.com', DB2_FENCED 'N');
```

Você deve especificar um endereço de e-mail quando registrar um wrapper. Este endereço de e-mail está incluído em todas as consultas e permite que o NCBI contacte-o se ocorrerem problemas, tais como, excessivas consultas sobrecarregando os servidores NCBI.

A próxima tarefa nesta seqüência de tarefas é definir a variável de ambiente `DB2_DJ_COMM` para o wrapper Entrez.

Related tasks:

- “Definindo a Variável de Perfil `DB2_DJ_COMM` do DB2 para o Wrapper Entrez” na página 283
- “Depois de Instalar Wrappers Não Relacionais” no *DB2 Information Integrator Installation Guide*

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper Entrez

A definição da variável de perfil DB2_DJ_COMM do DB2 para o wrapper Entrez faz parte da maior tarefa de adicionar Entrez a um sistema federado. Para aprimorar o desempenho quando origens de dados Entrez são acessadas, defina a variável de perfil DB2_DJ_COMM do DB2. Esta variável determina se o servidor federado carregará o wrapper na inicialização.

Procedimento:

Para definir a variável de perfil DB2_DJ_COMM do DB2, emita o comando **db2set** com a biblioteca de wrapper que corresponde ao wrapper especificado na instrução CREATE WRAPPER associada.

```
db2set DB2_DJ_COMM='libdb21sentrez.a'
```

Verifique se não existem espaços em nenhum lado do sinal de igual (=).

O uso do processador aumenta quando o servidor federado carrega as bibliotecas do wrapper durante a inicialização do banco de dados. Para evitar o uso excessivo, especifique apenas as bibliotecas que deseja acessar.

A próxima tarefa nesta seqüência de tarefas é registrar o servidor para uma origem de dados Entrez.

Related tasks:

- “Registrando o Servidor para uma Origem de Dados Entrez” na página 283

Registrando o Servidor para uma Origem de Dados Entrez

O registro do servidor para uma origem de dados Entrez faz parte da maior tarefa de adicionar Entrez a um sistema federado. Depois de registrar o wrapper, você deve registrar um servidor correspondente.

O banco de dados, PubMed ou Nucleotide, que é representado por uma origem de dados específica, é identificado pelo valor do tipo de servidor, conforme expresso na instrução CREATE SERVER. Este valor de tipo de servidor controla a estrutura de pseudônimos que são criados.

Procedimento:

Para registrar o servidor Entrez no sistema federado, emita uma instrução CREATE SERVER.

Por exemplo, para registrar um servidor denominado `pubmed_server1` para o wrapper `entrez_wrapper`, emita a seguinte instrução:

```
CREATE SERVER pubmed_server1
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper;
```

Além disso, para registrar um servidor denominado `nucleotid_server1` para o wrapper `entrez_wrapper`, emita a seguinte instrução:

```
CREATE SERVER nucleotid_server1
  TYPE NUCLEOTIDE
  VERSION 1.0
  WRAPPER entrez_wrapper;
```

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para origens de dados Entrez.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados Entrez” na página 284

Related reference:

- “Argumentos da Instrução CREATE SERVER - Wrapper Entrez” na página 382

Registrando Pseudônimos para Origens de Dados Entrez

O registro de pseudônimos para origens de dados Entrez faz parte da maior tarefa de adicionar um Entrez a um sistema federado.

Restrições:

O esquema para cada banco de dados Entrez é fixado pelo wrapper e não pode ser alterado nem corrigido. Para cada banco de dados, existe um conjunto de tabelas fixas com uma lista de colunas fixas para cada tabela. As tabelas em um banco de dados possuem um relacionamento hierárquico. Uma tabela, que é o pai de todas as outras tabelas no banco de dados, é chamada de tabela raiz. Todas as outras tabelas no banco de dados possuem um relacionamento entre pai/filho que retorna à tabela raiz.

Procedimento:

Para registrar pseudônimos para origens de dados Entrez, emita uma instrução CREATE NICKNAME.

Como a lista de colunas para os pseudônimos é fixada e fornecida pelo wrapper, a sintaxe básica para criar pseudônimos Nucleotide é simples. Por exemplo:

```

CREATE NICKNAME GBSeq FOR SERVER nuc1;
CREATE NICKNAME GBFeatures FOR SERVER nuc1;
CREATE NICKNAME GBIntervals FOR SERVER nuc1;
CREATE NICKNAME GBQualifiers FOR SERVER nuc1;
CREATE NICKNAME GBReference FOR SERVER nuc1;

```

Aqui está um exemplo da sintaxe básica para criar pseudônimos PubMed:

```

CREATE NICKNAME pmarticles FOR SERVER pubmed_server;
CREATE NICKNAME PMACCESSION FOR SERVER pubmed_server;
CREATE NICKNAME PMCHEMICAL FOR SERVER pubmed_server;
CREATE NICKNAME PMMESH FOR SERVER pubmed_server;
CREATE NICKNAME PMCOMMENTS FOR SERVER pubmed_server;
CREATE NICKNAME PMARTICLEID FOR SERVER pubmed_server;
CREATE NICKNAME PMURL FOR SERVER pubmed_server;

```

O nome do pseudônimo é o nome da tabela subjacente.

A utilização desta sintaxe limita você a uma família de pseudônimos por esquema DB2. Você pode utilizar outros nomes utilizando as opções de pseudônimo REMOTE_OBJECT e PARENT. Para um pseudônimo raiz, apenas REMOTE_OBJECT é requerido. Para qualquer outro pseudônimo, REMOTE_OBJECT e PARENT devem ser fornecidos.

O exemplo a seguir mostra o mesmo conjunto de pseudônimos Nucleotide utilizando a capacidade de renomeação:

```

CREATE NICKNAME NewSeq FOR SERVER nuc1 OPTIONS (REMOTE_OBJECT 'GBSEQ');
CREATE NICKNAME NewFeatures FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBFEATURES', PARENT 'NEWSEQ');
CREATE NICKNAME NewIntervals FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBINTERVALS', PARENT 'NEWFEATURES');
CREATE NICKNAME NewQualifiers FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBQUALIFIERS', PARENT 'NEWFEATURES');
CREATE NICKNAME NewReference FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBREFERENCE', PARENT 'NEWSEQ');

```

Este exemplo mostra o mesmo conjunto de pseudônimos PubMed utilizando a capacidade de renomeação:

```

CREATE NICKNAME newpmarticles FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMARTICLES');
CREATE NICKNAME NEWPMACCESSION FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMACCESSION', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMCHEMICAL FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMCHEMICAL', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMMESH FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMMESH', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMCOMMENTS FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMCOMMENTS', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMARTICLEID FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMARTICLEID', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMURL FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMURL', PARENT 'NEWPMARTICLES');

```

A próxima tarefa nesta seqüência de tarefas é registrar funções personalizadas para origens de dados Entrez.

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Tabelas de Esquemas PubMed e Nucleotide” na página 292

Funções Personalizadas e Consultas Entrez

O ambiente federado utiliza dois mecanismos de consulta. Para o wrapper Entrez, estes mecanismos de consulta são DB2 e Entrez. Com uma exceção, você especifica todos os predicados para o mecanismo Entrez através de funções personalizadas. Para o mecanismo DB2, especifique todos os predicados através de operadores relacionais.

A função personalizada principal é ENTREZ.CONTAINS. A função CONTAINS requer um argumento de coluna de termo de pesquisa e um argumento de texto de consulta. O exemplo a seguir mostra uma instrução ENTREZ.CONTAINS:

```
ENTREZ.CONTAINS (<search term column>, <query text>)
```

Uma marcação na coluna Q das tabelas de esquemas identifica um termo de pesquisa. O texto da consulta deve estar na sintaxe de consulta Entrez modificada. Esta sintaxe consiste em termos de pesquisa separados por operadores Booleanos (OR, AND e NOT) e agrupados utilizando parênteses. A sintaxe do argumento de texto da consulta CONTAINS se difere da sintaxe da consulta padrão Entrez no sentido de que os qualificadores de termo da pesquisa, tais como, [pd], não são permitidos.

AS funções personalizadas são registradas no esquema Entrez, que deve ser utilizado para referir-se às funções. Quando as funções personalizadas são utilizadas, seu valor de retorno deve ser comparado ao valor 1 em um predicado de igualdade.

Em algumas situações, os predicados DB2 e Entrez podem ser mistos, de forma que não podem ser conectados. Estes casos geram a mensagem de erro SQL0142N (“Instrução SQL não suportada”).

Por exemplo, na consulta a seguir, você não pode separar as partes do predicado que são processadas pelo wrapper (as chamadas ENTREZ.CONTAINS) e as partes que devem ser processadas pelo DB2 (o predicado relacional no BaseCountA).

```
WHERE  
  ENTREZ.CONTAINS (Organism, 'drosophila') = 1  
  OR (BaseCountA > 10 AND ENTREZ.CONTAINS (Keywords, 'glop') = 1)
```

Alguns campos de pesquisa não possuem colunas correspondentes no esquema Entrez. Por exemplo, no banco de dados nucleotide, o termo [ALL] pesquisa todos os campos, enquanto [WORD] pesquisa todo o texto livre associado a um registro. Pseudocolunas são fornecidas para estes termos de pesquisa. Se uma pseudocoluna for referida em uma lista de seleção, o valor NULL será retornado.

Você pode executar consultas que, de outra maneira, podem não ser permitidas, emitindo a função master ENTREZ.SEARCH_TERM. Se você especificar a função master ENTREZ.SEARCH_TERM, ela será a única função personalizada em uma consulta. Para cada consulta, pode haver somente uma função master ENTREZ.SEARCH_TERM por pseudônimo Entrez. Além disso, as funções SEARCH_TERM e CONTAINS não podem ser mistas para o mesmo pseudônimo na mesma consulta. O primeiro argumento, especificação da coluna, deve ser a coluna de chave primária para o pseudônimo pai. O segundo argumento, o texto da consulta, é um termo de pesquisa em formato Entrez que inclui qualificadores de campos de pesquisa. Este texto é transmitido sem ser modificado, exceto para escapes de URI conforme requeridos pela sintaxe de URI para o Entrez.

O exemplo a seguir mostra uma consulta com uma cláusula WHERE em um pseudônimo PubMed:

```
WHERE
  ENTREZ.CONTAINS (authors, 'kaufmann OR ito AND NOT rakesh')
  AND
  (ENTREZ.CONTAINS (title, 'drosophila')
OR
  ENTREZ.CONTAINS(alltext, 'drosophila OR "fruit fly"'))
```

Neste exemplo, os predicados individuais são authors, title e all text.

Os predicados individuais são modificados para que o qualificador seja adicionado após cada termo de pesquisa. Em seguida, os termos são agrupados com parênteses para forçar a precedência do operador Booleano do DB2. Por causa destas modificações, o predicado authors se torna:

```
((kaufmann[auth] OR ito[auth]) AND (NOT (rakesh[auth])))
```

O predicado title se torna:

```
(drosophila[tit1])
```

E o predicado all text se torna:

```
(drosophila[all] OR "fruit fly"[all])
```

Quando os predicados individuais são combinados, são utilizados parênteses para manter a precedência do operador Booleano do DB2. Excluindo

transformações de texto que são necessárias para expressar a cadeia como parte de um URI, a cadeia de termo de pesquisa final enviada para o Entrez é:

```
((kaufman[auth] OR ito[auth]) AND (NOT (rakesh[auth]))) AND  
((drosophila[tit1]) OR (drosophila[all] OR "fruit fly"[all]))
```

Related reference:

- “Tabela de Funções Personalizadas - Wrapper Entrez” na página 291

Predicados Relacionais para o Wrapper Entrez

O wrapper Entrez suporta predicados relacionais, tais como, =, BETWEEN, LIKE e <>, em colunas de pseudônimo. No entanto, o mecanismo de pesquisa Entrez processa apenas alguns destes predicados relacionais. Os predicados relacionais que não são processados pelo mecanismo de pesquisa Entrez são processados pelo DB2. O mecanismo de pesquisa Entrez processa os predicados de igualdade (=) e IN em determinadas colunas de ID para cada esquema. Estes predicados permitem que o wrapper Entrez ignore a fase de pesquisa e execute a fase de busca diretamente. Exemplos de predicados válidos são:

```
WHERE pmid = '1234567'  
WHERE medlineid IN ('1234567', '9191919')
```

As colunas que podem ser utilizadas neste tipo de predicado são identificadas pela coluna F das tabelas de esquemas. O valor desta opção deve ser Y.

Related concepts:

- “Cláusulas WHERE Inválidas para o Wrapper Entrez” na página 288

Related tasks:

- “Origem de Dados Entrez — Consultas de Exemplo” na página 290
- “Registrando Funções Personalizadas para o Wrapper Entrez” na página 281

Cláusulas WHERE Inválidas para o Wrapper Entrez

O wrapper Entrez rejeita qualquer consulta que resultará em uma varredura não qualificada do banco de dados NCBI. Uma cláusula WHERE válida deve conter um predicado de igualdade (ou IN) no ID primário para o esquema ou uma função personalizada. As consultas que não atendem estes critérios são rejeitadas com o código de erro SQL0142N ou SQL30090N.

Related concepts:

- “Predicados Relacionais para o Wrapper Entrez” na página 288

Related tasks:

- “Origem de Dados Entrez — Consultas de Exemplo” na página 290
- “Registrando Funções Personalizadas para o Wrapper Entrez” na página 281

Simplificação de Elementos de Dados de Esquema

Vários elementos de dados são convertidos em um formato canônico quando são apresentados através do esquema SQL. Estes elementos de dados incluem listas de itens, nomes e datas.

Listas de Itens

A menos que indicado de outra maneira, as listas de itens que não estão ajustados em uma única coluna possuem itens individuais separados por ponto e vírgula e por um único espaço. Por exemplo, se uma entrada contiver as palavras-chave `dnaA gene`, `dnaN gene` e `orf187`, a coluna de Palavras-chave correspondente conterá o valor `dnaA gene; dnaN gene; orf187`.

Nomes

Os nomes nos esquemas NCBI consistem em um sobrenome obrigatório e em um de vários elementos opcionais. Alguns destes elementos opcionais podem ocorrer juntos e outros são exclusivos entre si. Para criar um formato canônico de um nome, atribua uma precedência a estes elementos. Da ordem mais alta para a mais baixa, estes elementos são:

- Nome
- Nome ou nome do meio
- Iniciais

Você pode apresentar nomes com ou sem afiliações. Sem uma afiliação, um nome é formatado como `<last name>`, `<first>`, em que `<first>` é um dos elementos opcionais. Se o elemento `<first>` não for localizado, a vírgula não será utilizada. Uma afiliação pode ser adicionada no formato (`<affiliation>`).

Separe nomes em listas não ajustadas com um ponto e vírgula e um espaço. Um exemplo da forma correta de separar nomes é:

Parker, M. J.; Ranjan, K. A.

Datas

Datas, principalmente datas de publicação, possuem uma grande variedade de formatos nos esquemas NCBI. Para acomodar estes formatos e permitir comparações de datas e aritmética de data sempre que possível, as datas no esquema SQL são representadas em dois formatos. Primeiro, uma data pode ser uma cadeia de caracteres. Segundo, uma data pode ser uma coluna de tipo DATE.

Se apenas um mês estiver presente em um valor de data sem referência a um dia, o primeiro dia do mês será o dia padrão. Se uma estação estiver presente em vez de um mês, ou um mês e dia, o primeiro dia da estação será utilizado.

Origem de Dados Entrez — Consultas de Exemplo

Este tópico fornece algumas consultas de amostra a serem executadas em origens de dados Entrez.

Procedimento:

Para executar consultas, utilize os exemplos a seguir como um guia.

Em Pseudônimos PubMed:

A seguir é mostrada uma consulta com uma única chave de busca em um pseudônimo PubMed:

```
select PMID, ArticleTitle FROM pmarticles WHERE pmid = '12345';
```

A seguir é mostrada uma consulta com chaves de busca mistas em um pseudônimo PubMed:

```
select PMID, ArticleTitle FROM pmarticles  
WHERE pmid = '12345' OR MedlineID = '12346';
```

A seguir é mostrada uma consulta com uma função CONTAINS em um pseudônimo PubMed:

```
select PMID, ArticleTitle FROM pmarticles  
WHERE entrez.contains (ArticleTitle, 'granulation') = 1  
AND entrez.contains (PubDate, '1992') = 1;
```

A seguir é mostrada uma consulta que pesquisa o AuthorList e LanguageList especificados em um pseudônimo PubMed:

```
select PMID, ArticleTitle FROM pmarticles  
WHERE entrez.contains (AuthorList, 'Albarrak') = 1  
AND entrez.contains (LanguageList, 'eng')=1;
```

A seguir é mostrada uma consulta com um predicado complexo em um pseudônimo PubMed:

```
select PMID, ArticleTitle FROM pmarticles  
WHERE entrez.contains (PublicationTypeList, 'Journal Article') = 1  
AND entrez.contains (MedlineTA, 'sun')=1  
OR entrez.contains (PersonalNameSubjectList, 'shine')=1;
```

Em Pseudônimos Nucleotide:

A seguir é mostrada uma consulta com várias chaves de busca em um pseudônimo Nucleotide:

```
select PrimaryAccession, LocusName, SeqLength from gbseq
WHERE PrimaryAccession in ('NM_000890', 'NC_003106');
```

A seguir é mostrada uma consulta que pesquisa todos os campos pesquisáveis em um pseudônimo Nucleotide:

```
select PrimaryAccession, substr(Definition,1,300), GI from gbseq
WHERE entrez.contains(AllText, 'abcde')=1;
```

A seguir é mostrada uma consulta que pesquisa todos os textos livres em um pseudônimo Nucleotide:

```
select * from gbseq WHERE entrez.contains(FreeText, 'abcde')=1;
```

A seguir é mostrada uma consulta que pesquisa uma definição em um pseudônimo Nucleotide:

```
select PrimaryAccession, substr(Definition,1,300), version, GI from gbseq
WHERE entrez.contains(Definition, 'Sulfolobus tokodaii
AND complete genome') = 1;
```

A seguir é mostrada uma consulta que pesquisa uma palavra-chave em um pseudônimo Nucleotide:

```
select PrimaryAccession, substr(KeywordList,1,200), Segment from gbseq
WHERE entrez.contains(KeywordList, 'nkcc1 gene') = 1;
```

Related concepts:

- “Predicados Relacionais para o Wrapper Entrez” na página 288
- “Cláusulas WHERE Inválidas para o Wrapper Entrez” na página 288

Related tasks:

- “Registrando Funções Personalizadas para o Wrapper Entrez” na página 281

Tabela de Funções Personalizadas - Wrapper Entrez

Tabela 45. Funções personalizadas para o wrapper Entrez

| Nome da função | Descrição |
|--|---|
| CONTAINS (col VARCHAR(), term VARCHAR()), CONTAINS (col INTEGER, term VARCHAR()), CONTAINS (col SMALLINT, term VARCHAR()), CONTAINS (col REAL, term VARCHAR()), CONTAINS (col DOUBLE, term VARCHAR()), CONTAINS (col DATE, term VARCHAR()), CONTAINS (col TIME, term VARCHAR()), CONTAINS (col CHAR(), term VARCHAR()), CONTAINS (col TIMESTAMP(), term VARCHAR()) | <p>Pesquisa uma coluna com marcação utilizando a expressão especificada.</p> <p>col Coluna com marcação.</p> <p>term Termo de pesquisa.</p> |

Tabela 45. Funções personalizadas para o wrapper Entrez (continuação)

| Nome da função | Descrição |
|---|--|
| SEARCH_TERM (col VARCHAR(), term VARCHAR()) | Transmite um termo de pesquisa do Entrez diretamente para o mecanismo de pesquisa do Entrez. |
| | col Coluna com marcação. |
| | term Termo de pesquisa. |

Tabelas de Esquemas PubMed e Nucleotide

Este tópico fornece tabelas para os esquemas PubMed e Nucleotide.

Esquema PubMed

Este esquema define a aparência de dados de um servidor de tipo PubMed. O esquema consiste em vários pseudônimos relacionados. Nas tabelas a seguir, a coluna Q é a marcação do campo. Para obter uma lista de marcações pesquisáveis válidas, consulte <http://www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html#SearchFieldDescriptionsandTags>. A coluna F indica se a coluna de pseudônimo é uma chave de busca designada. A utilização de chaves de busca pode acelerar o processamento em alguns casos.

Tabela 46. Pseudônimo PubMed PMArticles

| Nome da coluna | Tipo de dados | Descrição | Q | F |
|----------------|----------------------|--|-----|---|
| PMID | VARCHAR(10) NOT NULL | PubMed ID PRIMARY_KEY Y | UID | S |
| MedlineID | VARCHAR(10) | Medline ID | UID | S |
| Proprietário | VARCHAR(8) NOT NULL | Proprietário da entrada de publicação; os valores são definidos por NCBI e podem ser NLM, NASA, PIP, KIE, HSR, HMD, SIS, NOTNLM. Se não estiver presente, o padrão será NLM. | | |

Tabela 46. Pseudônimo PubMed PMArticles (continuação)

| Nome da coluna | Tipo de dados | Descrição | Q | F |
|-----------------------------|-----------------------|--|------|---|
| Status | VARCHAR(32) NOT NULL | Status de publicação conforme definido por NCBI. Os valores podem incluir: In-Process, Completed, Out-of-scope, PubMed-not_MEDLINE | | |
| DateCreated | DATE NOT NULL | | | |
| DateCompleted | DATE | | | |
| DateRevised | DATE | | | |
| ArticleTitle | VARCHAR(250) NOT NULL | | TI | |
| Pagination | VARCHAR(32) | | | |
| Abstract | VARCHAR(32000) | | TIAB | |
| Affiliation | VARCHAR(250) | Afiliação e endereço do primeiro autor | AD | |
| AuthorList | VARCHAR(3200) | Lista de autores; canonizados | AU | |
| LanguageList | VARCHAR(250) NOT NULL | Lista separada por ponto e vírgula | LA | |
| PublicationTypeList | VARCHAR(250) NOT NULL | Lista separada por ponto e vírgula | PT | |
| VernacularTitle | VARCHAR(250) | | | |
| DateOfElectronicPublication | VARCHAR(32) | O esquema NCBI não especifica nenhuma estrutura para esta coluna | | |
| Country | VARCHAR(128) | | | |
| MedlineTA | VARCHAR(250) NOT NULL | | TA | |
| NlmUniqueid | VARCHAR(32) | Contém MedlineCode se NlmUniqueID não estiver presente | | |

Tabela 46. Pseudônimo PubMed PMArticles (continuação)

| Nome da coluna | Tipo de dados | Descrição | Q | F |
|-------------------------|---------------|---|----|-------|
| GeneSymbolList | VARCHAR(250) | Lista separada por ponto e vírgula; não utilizada desde 1996 | | |
| NumberOfReferences | INTEGER | | | |
| PersonalNameSubjectList | VARCHAR(250) | Canonizado como uma lista de nomes separada por ponto e vírgula | PS | |
| KeywordList | VARCHAR(3200) | Lista separada por ponto e vírgula | | |
| SpaceFlightMissionList | VARCHAR(250) | Lista separada por ponto e vírgula | | |
| InvestigatorList | VARCHAR(250) | Canonizado como uma lista de nomes separada por ponto e vírgula | | |
| PublicationStatus | VARCHAR(32) | | | |
| ProviderID | VARCHAR(32) | | | |
| CitationSubsetList | VARCHAR(250) | Lista separada por ponto e vírgula | SB | |
| AllFields | VARCHAR(1) | Pseudocoluna; sempre retorna NULL | | TODOS |
| TextWords | VARCHAR(1) | Pseudocoluna; sempre retorna NULL | | TW |
| PubDate | DATE | Inclui data de publicação de diário e de manual + data de linha mediana | | DP |
| PubDateString | VARCHAR(32) | Inclui data de publicação de diário e de manual + data de linha mediana | | DP |
| Título | VARCHAR(250) | Título do manual ou de diário | | TA |

Tabela 46. Pseudônimo PubMed PMArticles (continuação)

| Nome da coluna | Tipo de dados | Descrição | Q | F |
|-------------------------|---------------|--|----|---|
| Journal_ISSN | CHAR(9) | | TA | |
| Journal_Volume | VARCHAR(10) | | VI | |
| Journal_Issue | VARCHAR(10) | | IP | |
| Journal_Coden | VARCHAR(32) | | | |
| Journal_ISOAbbreviation | VARCHAR(32) | | | |
| Book_Publisher | VARCHAR(128) | | | |
| Book_Authors | VARCHAR(250) | Canonizado como outras listas de autores | | |
| Book_CollectionTitle | VARCHAR(128) | | | |
| Book_Volume | VARCHAR(10) | | | |

Tabela 47. Pseudônimo PubMed PMAccession

| Nome da coluna | Tipo de dados | Descrição | Q |
|----------------|-----------------------|------------------------|----|
| PMID | VARCHAR(10) NOT NULL | FOREIGN_KEY PMARTICLES | |
| DataBankName | VARCHAR(250) NOT NULL | | SI |
| Accession | VARCHAR(32) NOT NULL | | SI |

Tabela 48. Pseudônimo PubMed PMChemical

| Nome da coluna | Tipo de dados | Descrição | Q |
|-----------------|-----------------------|--|----|
| PMID | VARCHAR(10) NOT NULL | FOREIGN_KEY PMARTICLES | |
| NameOfSubstance | VARCHAR(128) NOT NULL | | NM |
| RegistryNumber | VARCHAR(32) NOT NULL | Pode ser CAS ou outro número de registro | RN |
| CASRegistry | CHAR | Y ou N | |

Tabela 49. Pseudônimo PubMed PMMeSHHeading

| Nome da coluna | Tipo de dados | Descrição | Q |
|--------------------|-----------------------|--|---|
| PMID | VARCHAR(10) NOT NULL | FOREIGN_KEY PMARTICLES | ID |
| DescriptorOrName | VARCHAR(128) NOT NULL | | MH (Se o predicado "DescriptorIs Major = Y" for incluído na consulta, o termo de pesquisa será MAJR.) |
| DescriptorIsMajor | CHAR NOT NULL | Y se o descritor for principal | |
| QualifierOrSubhead | VARCHAR(128) | | SH |
| QSIsmajor | CHAR | Y se o qualificador ou subtítulo for principal | |

Tabela 50. Pseudônimo PubMed PMComments

| Nome da coluna | Tipo de dados | Descrição | Q |
|----------------|-----------------------|---|---|
| PMID | VARCHAR(10) NOT NULL | FOREIGN_KEY PMARTICLES | |
| RefSource | VARCHAR(128) NOT NULL | | |
| Type | VARCHAR(32) NOT NULL | CommentOn, CommentIn, ErratumIn, ErratumFor, RepublishedFrom, RepublishedIn, RetractionOf, RetractionIn, UpdateIn, UpdateOf, SummaryForPatents, OriginalReportIn | |
| Note | VARCHAR(3200) | | |

Tabela 51. Pseudônimo PubMed PMArticleID

| Nome da coluna | Tipo de dados | Descrição | Q |
|----------------|----------------------|--|---|
| PMID | VARCHAR(10) NOT NULL | FOREIGN_KEY PMARTICLES | |
| ArticleID | VARCHAR(32) NOT NULL | | |
| IdType | VARCHAR(8) NOT NULL | doi, pii, pmcpid, pmpid, sici, pubmed, medline, pmcid | |

Tabela 52. Pseudônimo PubMed PMURL

| Nome da coluna | Tipo de dados | Descrição | Q |
|----------------|--------------------------|------------------------------------|---|
| PMID | VARCHAR(10) NOT NULL | FOREIGN_KEY PMARTICLES | |
| URL | VARCHAR(250) NOT NULL | | |
| Language | CHAR(2) | Código de Idioma ISO | |
| Type | CHAR(1) | F para FullText, S para Summary | |

Esquema Nucleotide

Consulte http://www.ncbi.nlm.nih.gov/entrez/query/static/help/Summary_Matrices.html#Search_Fields_and_Qualifiers.

Tabela 53. Pseudônimo Nucleotide GBSeq

| Nome da coluna | Tipo de dados | Descrição | Q | F |
|------------------|----------------------|---------------------------------|------|---|
| PrimaryAccession | VARCHAR(16) NOT NULL | Número de acesso primário | PACC | S |
| SequenceKey | VARCHAR(32) NOT NULL | PRIMARY_KEY Y | | |
| LocusName | VARCHAR(16) NOT NULL | | ACCN | |
| SeqLength | INTEGER NOT NULL | | SLEN | |

Tabela 53. Pseudônimo Nucleotide GBSeq (continuação)

| Nome da coluna | Tipo de dados | Descrição | Q | F |
|----------------|---------------------------|--|------|---|
| Strandedness | VARCHAR(32) | not-set, single- stranded, double- stranded, mixed- stranded | | |
| MoleculeType | VARCHAR(16) | nucleic-acid, dna, rna, trna, rrna, mrna, urna, snrna, snorna, peptide | PROP | |
| Topology | VARCHAR(16) | linear, circular | | |
| Division | CHAR(3) NOT NULL | | PROP | |
| UpdateDate | DATE NOT NULL | | MDAT | |
| CreateDate | DATE NOT NULL | | | |
| Definition | VARCHAR(7000) NOT NULL | | TITL | |
| Version | INTEGER | | | |
| GI | VARCHAR(16) | FETCH_KEY Y | UID | |
| KeywordList | VARCHAR(7000) | Lista separada por ponto e vírgula | KYWD | |
| Segment | VARCHAR(250) | | | |
| Source | VARCHAR(200) NOT NULL | | ORGN | |
| Organism | VARCHAR(7000) NOT NULL | | ORGN | |
| Taxonomy | VARCHAR(7000) NOT NULL | | | |
| Comment | VARCHAR(7000) | | | |
| Primary | VARCHAR(7000) | | | |
| SourceDB | VARCHAR(250) | | | |
| Sequence | CLOB | | | |

Tabela 53. Pseudônimo Nucleotide GBSeq (continuação)

| Nome da coluna | Tipo de dados | Descrição | Q | F |
|----------------|---------------|---|---|---|
| AllText | VARCHAR(1) | Pseudocoluna, TODOS sempre retorna NULL | | |
| FreeText | VARCHAR(1) | Pseudocoluna, WORD sempre retorna NULL | | |

Tabela 54. Pseudônimo GBReference

| Nome da coluna | Tipo de dados | Descrição | Q |
|----------------|-----------------------|---|------|
| SequenceKey | VARCHAR(32) NOT NULL | FOREIGN_KEY Y | |
| ReferenceNum | INTEGER NOT NULL | Analizador a partir de GBReference_reference | |
| RangeLow | INTEGER NOT NULL | Base inferior para referência (analisado a partir de GBReference_reference) | |
| RangeHigh | INTEGER NOT NULL | Base superior para referência (analisado a partir de GBReference_reference) | |
| Authors | VARCHAR(3200) | Lista de nomes separada por ponto e vírgula em GenBank form | AUTH |
| Consortium | VARCHAR(250) | | |
| Title | VARCHAR(250) | | WORD |
| Journal_Title | VARCHAR(250) NOT NULL | | JOUR |
| MedlineID | INTEGER | | |
| PubMedID | INTEGER | | |
| Remarks | VARCHAR(3200) | | |

Tabela 55. Pseudônimo Nucleotide GBFeatures

| Nome da coluna | Tipo de dados | Descrição | Q |
|-----------------|-----------------------|-------------------|------|
| SequenceKey | VARCHAR(32) NOT NULL | FOREIGN_KEY GBSEQ | |
| FeatureJoinKey | VARCHAR(32) NOT NULL | PRIMARY_KEY Y | |
| FeatureKey | VARCHAR(20) NOT NULL | | FKEY |
| FeatureLocation | VARCHAR(200) NOT NULL | | |

Tabela 56. Pseudônimo Nucleotide GBIntervals

| Nome da Coluna | Tipo de Dados | Descrição | Q |
|-------------------|----------------------|------------------------|---|
| FeatureJoinKey | VARCHAR(32) NOT NULL | FOREIGN_KEY GBFEATURES | |
| IntervalFrom | INTEGER | | |
| IntervalTo | INTEGER | | |
| IntervalPoint | INTEGER | | |
| IntervalAccession | VARCHAR(32) NOT NULL | | |

Tabela 57. Pseudônimo Nucleotide GBQualifiers

| Nome da coluna | Tipo de dados | Descrição | Q |
|----------------|----------------------|------------------------|---|
| FeatureJoinKey | VARCHAR(32) NOT NULL | FOREIGN_KEY GBFEATURES | |
| QualifierName | VARCHAR(50) | | |
| QualifierValue | VARCHAR(32000) | | |

Mensagens para o Wrapper Entrez

Este tópico descreve mensagens que podem ser encontradas ao trabalhar com o wrapper para Entrez. Para obter mensagens que não estão documentadas nesta tabela, no *Message Reference: Volume 1*, ou no *Message Reference: Volume 2*, entre em contato com o suporte de software IBM.

Tabela 58. Mensagens emitidas pelo wrapper para Entrez

| Código de Erro | Mensagem | Explicação |
|----------------|----------|------------|
|----------------|----------|------------|

Tabela 58. Mensagens emitidas pelo wrapper para Entrez (continuação)

| | | |
|----------|---|--|
| SQL0142N | A instrução SQL não é suportada. | Foi transmitido um tipo de consulta inválido para o wrapper. Verifique se a instrução SQL emitida é suportada por este wrapper. |
| SQL0204N | "<name>" é um nome indefinido. | O nome especificado é inválido. Verifique a instrução CREATE NICKNAME. |
| SQL0405N | O literal numérico "<literal>" não é válido porque seu valor está fora da faixa. | Uma coluna nos dados XML recuperados ou um predicado em uma instrução SQL contém um valor que está fora do possível intervalo para esse tipo de dados. Verifique se o tipo de dados para esta coluna e a coluna na origem de dados, ou redefina a coluna para um tipo mais apropriado. |
| SQL0408N | Um valor não é compatível com o tipo de dados do destino da atribuição. O nome de destino é "<nome_do_destino>". | Uma coluna nos dados XML contém caracteres que são inválidos para esse tipo de dados. Verifique se o tipo de dados para esta coluna e a coluna na origem de dados, ou redefina a coluna para um tipo mais apropriado. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Impossível localizar protótipo do banco de dados.") | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Não há dados a serem descompactados.") | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |

Tabela 58. Mensagens emitidas pelo wrapper para Entrez (continuação)

| | | |
|----------|--|--|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Erro ao criar o objeto do wrapper.") | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Tipo de expressão inválido.") | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Impossível localizar pseudônimo.") | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão "Erro de alocação de memória.") | Não há memória suficiente para processar o pedido de alocação dentro do wrapper. |
| SQL1816N | O wrapper "<wrapper_name>" não pode ser utilizado para acessar a "versão" da origem de dados (" <server_type>", "<server_version>") que você está tentando definir para o servidor federado.</server_type> | Um valor na cláusula VERSION da instrução CREATE SERVER é inválido. |
| SQL1816N | O wrapper "<wrapper_name>" não pode ser utilizado para acessar o "tipo" de origem de dados (" <server_type>", "<server_version>") que você está tentando definir para o servidor federado.</server_type> | Um valor na cláusula TYPE da instrução CREATE SERVER é inválido. |

Tabela 58. Mensagens emitidas pelo wrapper para Entrez (continuação)

| | | |
|-----------|--|---|
| SQL1817N | A instrução CREATE SERVER não identifica o "tipo" de origem de dados que você deseja definir para o banco de dados federado. | A cláusula TYPE da instrução CREATE SERVER é requerida mas não foi especificada. |
| SQL1822N | Código de erro inesperado "900" recebido da origem de dados "Wrapper Entrez." Texto e tokens associados são "Pseudônimo pai não definido." | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |
| SQL1823N | Não existe mapeamento de tipo de dados para o tipo de dados "<data_type>" do servidor "<server_name>." | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |
| SQL1881N | "<option_name>" não é um "<option_type>" válido para "<option_name>." | A opção especificada não é uma opção válida. Verifique a instrução CREATE NICKNAME. |
| SQL1882N | A opção "<option_type>" "<option_name>" não pode ser definida como "<option_value>" para "<option_name>." | O valor especificado não é válido para esta opção. Verifique a instrução CREATE NICKNAME. |
| SQL1883N | "<option_name>" é uma opção "<option_type>" requerida para "<option_name>." | A opção especificada é requerida para o objeto, mas não foi especificada. Verifique a instrução CREATE NICKNAME. |
| SQL1884N | Você especificou "FOREIGN_KEY" (uma opção "COLUMN") mais de uma vez. | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |
| SQL1884N | Você especificou "PRIMARY_KEY" (uma opção "COLUMN") mais de uma vez. | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Impossível alterar versão do servidor". | A versão de um servidor não pode ser alterada, emitindo a instrução ALTER SERVER. Um novo servidor deve ser criado com a nova versão. |

Tabela 58. Mensagens emitidas pelo wrapper para Entrez (continuação)

| | | |
|-----------|---|---|
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Pseudônimo PARENT inválido". | O pseudônimo referido em uma opção de pseudônimo PARENT não é válido para o pseudônimo atual. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Nome de coluna inválido". | Um nome de coluna especificado na instrução CREATE NICKNAME não corresponde a nenhuma das possíveis colunas para o pseudônimo. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Impossível buscar chaves AND". | Várias referências a uma chave de busca, como a coluna PMID do pseudônimo PMArticles, foram criadas em uma junção. Por exemplo, "PMID = 12346 AND PMID = 12348". Os predicados de chave de busca podem ser associados apenas utilizando OR. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Funções SEARCH_TERM e CONTAINS mistas". | As funções SEARCH_TERM e CONTAINS não podem ser mistas em uma consulta. É permitida apenas uma função SEARCH_TERM por consulta. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Primeiro argumento inválido na função". | O primeiro argumento em uma função SEARCH_TERM ou CONTAINS era inválido. Este argumento deve ser uma referência a uma coluna. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Segundo argumento inválido na função". | O segundo argumento em uma função SEARCH_TERM ou CONTAINS era inválido. Este argumento deve ser um literal de cadeia, uma variável de host ou uma referência de coluna. |

Tabela 58. Mensagens emitidas pelo wrapper para Entrez (continuação)

| | | |
|-----------|--|---|
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Coluna sem marcação na função CONTAINS". | O primeiro argumento na função CONTAINS era inválido. Este argumento deve ser uma referência a uma coluna com marcação. |
| SQL30090N | Operação inválida para o ambiente de execução do aplicativo. Código de razão = "Função inválida". | Este é um erro interno. Entre em contato com o Suporte de Software da IBM. |

Capítulo 18. Configurando o Acesso às Origens de Dados do Extended Search

Este capítulo explica o que é o IBM Lotus Extended Search, como incluir origens de dados do Extended Search em seu sistema federado e como utilizar SQL para pesquisar as origens de dados do Extended Search. Este capítulo lista também as mensagens de erro associadas ao wrapper do Extended Search.

O que É Extended Search?

O produto Extended Search é um sistema cliente/servidor multicamada que fornece amplos recursos de pesquisa e recuperação. Com o Extended Search, você pode inserir um único pedido e pesquisar milhares de possíveis repositórios de dados e a Internet ao mesmo tempo. Estes repositórios, que podem ter conteúdo e estrutura variados, podem ser geograficamente dispersos por todo o mundo.

O Extended Search suporta pesquisa distribuída e heterogênea de dados estruturados e não estruturados através de um único ponto de acesso. Ele alavanca seu investimento atual de gerenciamento de dados e trata totalmente a logística requerida para acessar muitas origens diferentes ao mesmo tempo.

O Extended Search utiliza seu GQL (generalized query language) como uma sintaxe de pesquisa e, internamente, converte cada pedido de pesquisa nas linguagens de pesquisa nativas das origens de dados que você deseja pesquisar. Ele também utiliza métodos que são nativos para as origens localizarem e recuperarem informações sem considerar a localização de uma origem.

Consulte a documentação do produto Extended Search para obter informações sobre como instalar um servidor Extended Search, configurar o domínio de pesquisa e utilizar a GQL. Os documentos a seguir estão disponíveis na página Resources do Web site do IBM® Lotus® Extended Search:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

Extended Search General Information

Descreve os componentes em um sistema Extended Search e como eles interagem entre si e os sistemas de dados de backend.

Extended Search Installation

Define os pré-requisitos do sistema e fornece instruções para instalar o produto e verificar o processo de instalação.

Extended Search Administration

Fornece instruções para adicionar origens de dados ao domínio de pesquisa, configurar campos pesquisáveis e utilizar aplicativos de pesquisa de amostra para consultar origens do Extended Search.

Extended Search Programming

Discute as ferramentas de desenvolvimento de aplicativos que podem ser utilizadas para estender o suporte de pesquisa para origens de dados que não são suportadas na configuração padrão do produto. Inclui uma descrição do GQL do Extended Search.

Origens de Dados Extended Search

Com o Extended Search, você pode pesquisar os seguintes tipos de origens de dados:

- Muitos Web sites de pesquisa e sites de notícias populares. Se desejar pesquisar o site de pesquisa da intranet ou outros sites de pesquisa internos ou externos, você poderá facilmente adicionar suporte para isso.
- Sistemas de correio, como os gerenciados com o Lotus Notes® e Microsoft® Exchange Server.
- Sistemas de gerenciamento de documentos, como DB2® Information Integrator para Bancos de Dados de Conteúdo.
- Bancos de dados relacionais, como IBM DB2, Oracle, Microsoft SQL Server, Microsoft Access e outros bancos de dados que estão de acordo com padrões ODBC (Open Database Connectivity).
- Índices de texto completos, como os que são criados com o IBM WebSphere® Portal, Domino™ Domain Index, Microsoft Index Server e Microsoft Site Server.
- Repositórios do Lotus, incluindo bancos de dados do Notes, bibliotecas e gabinetes Domino.Doc, locais do Lotus QuickPlace™ e K-maps (Lotus Discovery Server knowledge maps).
- Sistemas de mensagens instantâneas, como Lotus Sametime. Este recurso permite direcionar consultas para pessoas capacitadas, não apenas repositórios de dados pesquisáveis.
- Diretórios LDAP (Lightweight Directory Access Protocol), como os que são gerenciados com o IBM SecureWay, Domino LDAP Server e Exchange LDAP Server.
- Sistemas de arquivos. Você pode adicionar arquivos de texto de pesquisa que são armazenados localmente ou em unidades de rede. Você não pode pesquisar arquivos compactados ou criptografados.

Com APIs (interfaces de programação de aplicativos) do Extended Search C++ e Java™, você pode estender o suporte para outros tipos de origens, como bancos de dados de propriedade que não são mencionados aqui.

Como Funciona o Wrapper Extended Search

Em um modelo de banco de dados relacional estruturado, as colunas são denominadas e representadas em um formato consistente. Este recurso permite executar operações de cálculos precisas e unir dados de diferentes tabelas, comparando valores de colunas específicos. Você também pode fazer outros tipos de análise, como listar objetos em uma tabela que estão ausentes em outra tabela.

Em contrapartida, os dados não estruturados geralmente são armazenados em formato de texto livre. Geralmente, não existe nenhum ou poucos metadados que permitem consultar informações por nome de coluna. Uma pesquisa de dados não estruturados depende mais da localização de dados correspondentes a palavras-chave especificadas pelo usuário do que de critérios de cálculos.

O wrapper Extended Search combina estas duas técnicas de pesquisa. Com o wrapper, você pode utilizar SQL para pesquisar conteúdo não estruturado em um domínio do Extended Search. Você pode então executar operações analíticas ou relacionais nos resultados da pesquisa.

Emita consultas, inserindo as instruções SQL que se referem a uma tabela do DB2 com finalidade especial (uma tabela de pseudônimos). O Extended Search executa a pesquisa de acordo com os critérios de SQL e ocupa a tabela de pseudônimos com dados do resultado. Como os resultados da pesquisa persistem em uma tabela, os dados estão disponíveis para operações com outras tabelas do banco de dados, incluindo outras tabelas de pseudônimos.

Ao enviar um pedido de pesquisa com o wrapper, você pode recuperar dados de qualquer origem do Extended Search que sejam mapeados para uma tabela de pseudônimos. Você pode integrar estes dados com outras origens de dados em seu sistema federado, sem mover os dados para fora da origem de dados nativa. Os resultados da pesquisa aparecem como um único conjunto de resultados, independentemente de quantas origens fornecem respostas para a consulta.

A figura a seguir mostra como o wrapper Extended Search conecta diversas origens de dados em um domínio do Extended Search a um sistema federado. O wrapper acessa e recupera dados de um ou mais servidores remotos Extended Search. Se o wrapper contactar um servidor Extended Search que esteja conectado a outros servidores Extended Search, os resultados da pesquisa poderão ser retornados de vários servidores.

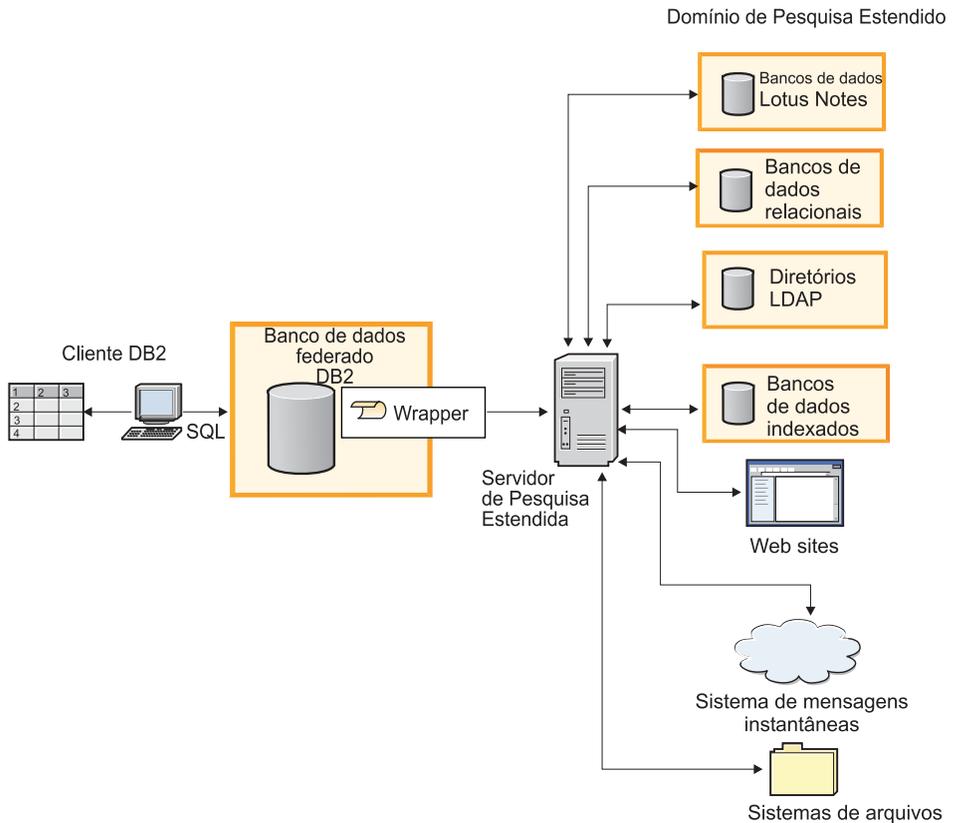


Figura 13. Como funciona o wrapper Extended Search

Related tasks:

- “Adicionando Origens de Dados Extended Search a um Servidor Federado” na página 315

Pseudônimos do Extended Search

No modelo de dados do Extended Search, um ou mais campos constituem um documento. Uma coleta de documentos constitui uma origem de dados. Você pode combinar qualquer número de origens de dados em uma categoria, que permite pesquisá-las e administrá-las como um grupo.

Para assegurar que usuários acessem apenas as origens de dados das quais eles precisam, uma categoria deve pertencer a, pelo menos, um aplicativo. Considere aplicativos como uma forma de agrupar usuários para finalidades de controlar os recursos de acesso e de pesquisa. Por exemplo, um aplicativo pessoal pode incluir as mesmas origens de dados que um aplicativo

financeiro, mas os usuários de cada aplicativo não precisariam, necessariamente, acessar os mesmos campos nessas origens de dados.

Ao registrar pseudônimos, você identifica os aplicativos, categorias, origens de dados e campos de origens de dados que deseja pesquisar. Estas entidades devem existir no banco de dados de configuração do Extended Search. Para pesquisar uma origem de dados Extended Search com o wrapper Extended Search, é necessário criar um pseudônimo para a origem.

O conteúdo da tabela de pseudônimos reflete o estado do banco de dados de configuração do Extended Search no momento que você registra o pseudônimo. Se um administrador do Extended Search atualizar a configuração (por exemplo, adicionando ou excluindo origens ou campos), essas alterações não serão refletidas na tabela de pseudônimos. Se uma tabela de pseudônimos referir-se a dados alterados e você desejar permanecer atualizado com o banco de dados de configuração do Extended Search, deverá alterar o pseudônimo ou eliminá-lo e criar um novo pseudônimo.

Se você não alterar ou recriar o pseudônimo, poderá receber erros e relatórios sem resultados quando tentar pesquisar itens que não mais existem no domínio do Extended Search.

Embora uma única tabela de pseudônimos possa conter informações sobre todas as origens que estão configuradas no Extended Search, a criação de várias tabelas de pseudônimos pode ser mais útil. Para utilizar toda a capacidade do DB2, crie um pseudônimo separado para cada tipo de origem de dados que planeja pesquisar com o wrapper Extended Search.

Por exemplo, você pode ter um pseudônimo para origens da Web, um para bancos de dados Notes, um para sistemas de arquivos, etc. Tendo tabelas de pseudônimos separadas, você poderá executar melhor as uniões nos dados que são retornados ao wrapper, relatar diversas origens com base nos valores de campos e integrar os dados de resultados a outros dados no sistema federado.

Related concepts:

- “Tabelas Verticais do Extended Search” na página 312

Related tasks:

- “Registrando Pseudônimos para Origens de Dados Extended Search” na página 317

Related reference:

- “Wrapper Extended Search - Consultas de Exemplo” na página 323

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper Extended Search” na página 366

Tabelas Verticais do Extended Search

Um aplicativo do Extended Search pode consistir em várias categorias que, por sua vez, podem conter muitas origens de dados. Como cada origem de dados utiliza suas próprias convenções para nomes de campos, uma interseção de campos pode resultar em um conjunto vazio. Ao mapear campos da origem de dados para colunas definidas pelo usuário em tabelas de pseudônimos e apresentar resultados da pesquisa como uma tabela horizontal, a tabela pode conter um número de colunas não gerenciável. Se muitas linhas contiverem apenas algumas colunas com dados, a tabela aparecerá esparsamente populada. Por exemplo:

| Coluna_1 | Coluna_2 | Coluna_3 |
|----------|----------|----------|
| Valor_11 | | |
| | Valor_22 | |
| Valor_31 | | Valor_33 |

No Extended Search, você pode controlar a apresentação de resultados, definindo campos mapeados. Os campos mapeados fornecem uma forma de combinar conteúdo que possui uma finalidade comum, mas que é denominado de forma diferente em origens diferentes. Por exemplo, você pode criar um campo mapeado denominado EmployeeNumber para representar dados de resultados de campos que são denominados EmpNum, EmpNo e EmpID em várias origens. Sem este recurso de mapeamento, é necessário definir uma coluna de pseudônimo para cada nome de campo exclusivo, em vez de para uma única coluna para o campo mapeado.

O mapeamento de campos é útil quando você sabe os nomes dos campos que precisam ser relatados. No entanto, alguns aplicativos precisam relatar um grande número de campos a partir de muitas origens de dados. O relacionamento entre os campos, principalmente para campos não estruturados, pode não ser conhecido antecipadamente. Assim, fica difícil definir e estruturar tabelas de pseudônimos significativas. Para suportar este tipo de aplicativo, o wrapper Extended Search permite criar uma tabela de pseudônimos vertical.

Ao criar uma tabela de pseudônimos para o Extended Search, você pode ativar a opção VERTICAL_TABLE. Esta opção retorna todos os campos que estão configurados para serem retornados em uma origem de dados, conforme definido no banco de dados de configuração do Extended Search. Utilize esta opção quando não tiver certeza sobre quais colunas serão relevantes em sua

pesquisa ou quais colunas serão relevantes quando executar consultas de pós-processamento ou uniões nos conjuntos de resultados.

Cada linha na tabela vertical contém informações sobre um campo que foi retornado no conjunto de resultados. Para cada linha, o Extended Search retorna o nome da origem do campo, o nome do campo, seu valor e seu tipo de dados (data, inteiro e outros). Diferente de resultados que são dispersos em colunas em uma tabela horizontal, a tabela vertical já está densamente populada e contém muitas linhas de dados. Por exemplo:

| Field_Name | Field_Value | Field_Datatype |
|------------|-------------|----------------|
| Coluna_1 | Valor_11 | VARCHAR |
| Coluna_2 | Valor_22 | DATE |
| Coluna_1 | Valor_31 | VARCHAR |
| Coluna_3 | Valor_33 | VARCHAR |

Você pode executar operações SQL nestes dados quando consultar a tabela e poderá consultar todos os rótulos da coluna. Por exemplo:

```
Field_Value LIKE '%IBM%'
```

Como a opção VERTICAL_TABLE retorna informações sobre todos os campos retornáveis em uma origem de dados, não será preciso consultar colunas específicas definidas pelo usuário. Se você ativar esta opção e, em seguida, emitir uma instrução SELECT para pesquisar colunas definidas pelo usuário, poderá receber informações duplicadas nos resultados da pesquisa. No entanto, se definir colunas definidas pelo usuário, você poderá utilizar essas colunas em uniões com outras tabelas no sistema federado.

A tabela a seguir resume as colunas fornecidas pelo sistema que o Extended Search retorna para cada linha em uma tabela de pseudônimos vertical.

| Nome da Coluna | Tipo de Dados | Descrição |
|---|---------------|---|
| <i>O wrapper sempre retorna as três seguintes colunas fixas para cada pseudônimo.</i> | | |
| DOC_ID | VARCHAR(512) | O identificador de documento, exclusivo para cada item em um conjunto de resultados da pesquisa. |
| DOC_RANK | INTEGER | A posição de relevância do documento. |
| CLIENT_LOCALE | VARCHAR(5) | O locale do cliente do pedido de pesquisa. Se a consulta SQL não fornecer o locale do cliente, a consulta utilizará enUS como o locale padrão do cliente. |

O wrapper cria as seguintes colunas fixas apenas se a opção VERTICAL_TABLE estiver ativada.

| Nome da Coluna | Tipo de Dados | Descrição |
|-----------------|---------------|--|
| DATASOURCE_NAME | VARCHAR(128) | O nome da origem de dados que produziu o resultado da pesquisa. |
| FIELD_NAME | VARCHAR(128) | O nome de um campo que foi retornado no resultado da pesquisa. |
| FIELD_VALUE | VARCHAR(4096) | O valor de um campo que foi retornado em um conjunto de resultados. Se o valor do campo for maior que o comprimento máximo da coluna do pseudônimo (o valor VARCHAR), o valor do campo será truncado. O token ES_TRUNCATE no final da coluna indica que o valor está incompleto. |
| FIELD_DATATYPE | SMALLINT | Um valor inteiro que representa o tipo de dados real do valor do campo: 384 DATE 448 VARCHAR 484 DECIMAL 496 INTEGER |

Uma tabela vertical, que armazena dados de resultado como valores VARCHAR, pode ser de difícil consulta. Para uma pesquisa mais precisa, crie campos mapeados no banco de dados de configuração do Extended Search e, em seguida, defina-os na tabela de pseudônimos. Com campos mapeados, você pode criar uma tabela de resultados da pesquisa horizontal concisa. Você também pode otimizar sua capacidade para executar operações relacionais nos resultados e combiná-las em consultas que envolvem outras tabelas no sistema do banco de dados federado.

Para obter informações sobre como definir campos mapeados no Extended Search, consulte o *Extended Search Administration*, que está disponível na página Resources do Web site do IBM® Lotus® Extended Search:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

Related concepts:

- “Pseudônimos do Extended Search” na página 310

Related tasks:

- “Registrando Pseudônimos para Origens de Dados Extended Search” na página 317

Related reference:

- “Wrapper Extended Search - Consultas de Exemplo” na página 323

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper Extended Search” na página 366

Adicionando Origens de Dados Extended Search a um Servidor Federado

Você pode instalar o wrapper Extended Search nos sistemas operacionais Microsoft Windows NT, Microsoft Windows 2000 e IBM AIX. Pode utilizar o wrapper para pesquisar servidores Extended Search existentes nos sistemas operacionais Windows, AIX, Sun Solaris e Red Hat Linux para Intel.

Pré-requisitos:

Antes de utilizar o wrapper Extended Search, assegure que as origens que você planeja pesquisar estejam configuradas no banco de dados de configuração do Extended Search. Envie algumas consultas através do cliente do Extended Search para verificar sua capacidade de pesquisar as origens antes de tentar pesquisá-las com o wrapper Extended Search.

Procedimento:

Para adicionar origens de dados Extended Search a um sistema federado:

1. Registre o wrapper Extended Search.
2. Registre o wrapper Extended Search que deseja utilizar com o wrapper.
3. Registre pseudônimos para origens Extended Search que deseja pesquisar.
4. Registre mapeamentos de usuários para usuários do DB2 que precisam pesquisar origens Extended Search.
5. Registre a função personalizada do Extended Search para especificar opções de pesquisa.

Related tasks:

- “Registrando o Wrapper Extended Search” na página 316
- “Registrando o Servidor para Origens de Dados Extended Search” na página 316
- “Registrando Pseudônimos para Origens de Dados Extended Search” na página 317
- “Registrando Mapeamentos de Usuários para Origens de Dados Extended Search” na página 318
- “Registrando a Função Personalizada do Extended Search” na página 320

Registrando o Wrapper Extended Search

Esta tarefa é parte da tarefa principal para inclusão de origens de dados Extended Search em um sistema federado. Para pesquisar origens de dados Extended Search, primeiro você deve registrar o wrapper Extended Search.

Procedimento:

Para registrar um wrapper Extended Search, emita uma instrução CREATE WRAPPER a partir do Processador da Linha de Comandos do DB2.

Por exemplo, para registrar um wrapper denominado NotesDBwrapper em um sistema Windows, emita a seguinte instrução:

```
CREATE WRAPPER NotesDBwrapper LIBRARY 'db2uies.dll'
```

Para registrar um wrapper denominado myESwrapper em um sistema AIX, emita a seguinte instrução:

```
CREATE WRAPPER myESwrapper LIBRARY 'libdb2uies.a'
```

A próxima tarefa nesta seqüência de tarefas é registrar o servidor para origens de dados Extended Search.

Related tasks:

- “Registrando o Servidor para Origens de Dados Extended Search” na página 316

Related reference:

- “Sintaxe da Instrução CREATE WRAPPER - Wrapper Extended Search” na página 386

Registrando o Servidor para Origens de Dados Extended Search

Esta tarefa é parte da tarefa principal para inclusão de origens de dados Extended Search em um sistema federado. Depois de registrar um wrapper, você deve criar uma definição de servidor correspondente para identificar o servidor remoto Extended Search que está sendo integrado ao sistema federado. Esta definição permite que o wrapper conecte-se ao servidor Extended Search.

Procedimento:

Para registrar um pseudônimo do Extended Search, emita uma instrução CREATE SERVER a partir do Processador da Linha de Comandos do DB2.

Por exemplo, para registrar um servidor denominado es1 para um wrapper denominado myESwrapper, emita a seguinte instrução. O servidor Extended Search utiliza o valor da porta padrão.

```
CREATE SERVER es1 WRAPPER myESwrapper OPTIONS (ES_HOST 'my.server.com')
```

Para criar este mesmo servidor, ative o rastreo para todos os níveis de mensagens (crítico, não crítico, aviso e informativo) e grave as mensagens de rastreo em um arquivo denominado es1wrapper.log; no diretório wrapper, emita a seguinte instrução:

```
CREATE SERVER es1 WRAPPER myESwrapper OPTIONS (ES_HOST 'my.server.com',  
        ES_TRACING 'ON', ES_TRACELEVEL 'CNWI',  
        ESTRACEFILENAME '/wrapper/es1wrapper.log')
```

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para origens de dados Extended Search.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados Extended Search” na página 317

Related reference:

- “Sintaxe da Instrução CREATE SERVER - Wrapper Extended Search” na página 383

Registrando Pseudônimos para Origens de Dados Extended Search

Esta tarefa é parte da tarefa principal para inclusão de origens de dados Extended Search em um sistema federado. Depois de registrar um servidor, é necessário registrar pelo menos um pseudônimo. Uma tabela de pseudônimos é uma tabela virtual do DB2 que identifica uma ou mais origens pesquisáveis em um domínio do Extended Search. Ao enviar uma consulta, especifique o pseudônimo para as origens que deseja pesquisar.

Pré-requisitos:

Certifique-se de que o servidor Extended Search para o qual estão sendo criados pseudônimos esteja em execução. Ao criar um pseudônimo, o sistema verifica se as informações sobre as origens e campos que você planeja pesquisar existem no banco de dados de configuração do Extended Search.

Procedimento:

Para registrar um pseudônimo do Extended Search, emita uma instrução CREATE NICKNAME a partir do Processador da Linha de Comandos do DB2.

Por exemplo, emita a seguinte instrução para criar uma tabela de pseudônimos para pesquisar todas as origens de dados que pertencem à categoria da Web no aplicativo Demo que é hospedado pelo servidor es1 Extended Search. Retorne aos campos WebTitle e WebDescription e utilize as opções de processamento de pesquisa padrão.

```
CREATE NICKNAME allweb (WebTitle VARCHAR(255), WebDescription VARCHAR(1000))
  FOR SERVER es1 OPTIONS(APPLICATIONID 'Demo', CATEGORY 'Web')
```

Emita a seguinte instrução para criar uma tabela de pseudônimos para pesquisar diversas origens de dados no aplicativo Science. Apresente os resultados da pesquisa como uma lista vertical de nomes de colunas, defina o valor de tempo limite como 60 segundos, permita que cada origem retorna até 100 documentos de resultados, expanda o tamanho do conjunto de resultados para 1000 entradas e classifique os resultados por nome do autor.

```
CREATE NICKNAME stars (Title VARCHAR(80), Author VARCHAR(40),
  Abstract VARCHAR(200))
  FOR SERVER es1 OPTIONS (APPLICATIONID 'Science',
  DATASOURCES 'Astronomy;NASA Library;Astrophysics', VERTICAL_TABLE 'yes',
  TIMEOUT '60', MAXHITS '100', TOTALMAXHITS '1000', SORTFIELD 'Author')
```

A próxima tarefa nesta seqüência de tarefas é registrar mapeamentos de usuários para o wrapper Extended Search.

Related concepts:

- “Pseudônimos do Extended Search” na página 310
- “Tabelas Verticais do Extended Search” na página 312

Related tasks:

- “Registrando Mapeamentos de Usuários para Origens de Dados Extended Search” na página 318

Related reference:

- “Wrapper Extended Search - Consultas de Exemplo” na página 323
- “Sintaxe da Instrução CREATE NICKNAME - Wrapper Extended Search” na página 366

Registrando Mapeamentos de Usuários para Origens de Dados Extended Search

Esta tarefa é uma etapa opcional na principal tarefa de adicionar origens de dados Extended Search a um sistema federado.

Os mapeamentos de usuários fornecem uma maneira de autenticar o acesso de usuários que consultam uma origem do Extended Search com o wrapper Extended Search. Se um usuário enviar uma consulta SQL a um pseudônimo

registrado do Extended Search e nenhum mapeamento de usuário estiver definido para esse usuário, o wrapper Extended Search utilizará um ID do usuário e senha padrão na tentativa de recuperar dados do servidor remoto Extended Search. Se a origem de dados que estiver sendo consultada exigir autenticação, poderá ser retornado um conjunto de resultados vazio.

Para assegurar que o ID do usuário e senha corretos tenham sido transmitidos para o servidor Extended Search, crie mapeamentos de usuários no sistema federado para usuários que estão autorizados a pesquisar origens do Extended Search. Quando você cria um mapeamento de usuário, a senha é armazenada em formato criptografado em uma tabela de catálogo do DB2. A senha permanece em formato seguro conforme é transmitida do DB2 ao Extended Search para as origens que estão sendo pesquisadas.

As definições de segurança no banco de dados de configuração do Extended Search determinam se o ID do usuário e senha estão autorizados a acessar as origens que estão sendo pesquisadas e se algum mapeamento adicional do ID do usuário será executado.

Procedimento:

Para registrar mapeamentos de usuários do Extended Search, emita uma instrução CREATE USER MAPPING a partir do Processador da Linha de Comandos do DB2.

A instrução deve identificar o ID do usuário do DB2 que precisa ser mapeado, o servidor Extended Search que hospeda origens de dados de destino e o ID do usuário e senha que permitem que o usuário acesse essas origens de dados.

Por exemplo, a instrução a seguir registra o ID do usuário user1 para que ele possa utilizar o servidor Extended Search es1 para pesquisar bancos de dados remotos.

```
CREATE USER MAPPING FOR user1 SERVER es1 OPTIONS  
  (REMOTE_AUTHID 'ESUserId', REMOTE_PASSWORD 'abc123def')
```

A próxima tarefa nesta seqüência de tarefas é registrar o gabarito de função personalizada do Extended Search.

Related tasks:

- “Registrando a Função Personalizada do Extended Search” na página 320

Related reference:

- “Sintaxe da Instrução CREATE USER MAPPING - Wrapper Extended Search” na página 385

Registrando a Função Personalizada do Extended Search

Esta tarefa é uma etapa opcional na principal tarefa de adicionar origens de dados Extended Search a um sistema federado.

As funções personalizadas não contêm código executável. Depois de registrar uma função, você pode referir-se a ela em consultas para alterar o comportamento padrão da pesquisa. A função personalizada para o wrapper Extended Search, `ES_SEARCH`, permite especificar expressões de pesquisa exatas e conteúdo de pesquisa que não estejam definidos como uma coluna na tabela de pseudônimos.

Restrições:

- Você pode chamar a função `ES_SEARCH` apenas com uma cláusula `WHERE`.
- A cláusula `WHERE` deve conter, pelo menos, um predicado que sirva como um predicado de pesquisa, a função `ES_SEARCH` ou um predicado de tipo "column-name operator constant."
- A função `ES_SEARCH` é um gabarito de função escalar. Ela deve utilizar o operador `EQUAL (=)` e o valor de comparação deve ser um (1).
- O primeiro parâmetro na função `ES_SEARCH` serve como um valor de âncora para identificar o pseudônimo ao qual a função deve ser aplicada, tal como, a posição do documento (`DOC_RANK`) nos resultados da pesquisa. Você deve especificar um campo `INTEGER` para este parâmetro. Este parâmetro, que não é avaliado, é muito importante se a consulta SQL contiver mais de um pseudônimo ou uma combinação de pseudônimos e tabelas. Por exemplo:

```
SELECT * FROM ES_N1, ES_N2
WHERE ESWRAPPER.ES_SEARCH(ES_N1.DOC_RANK, 'IBM') = 1 AND
      ESWRAPPER.ES_SEARCH(ES_N2.DOC_RANK, 'IBM') = 1
```

Procedimento:

Para registrar a função personalizada do Extended Search, emita a seguinte instrução `CREATE FUNCTION` a partir do Processador da Linha de Comandos do DB2:

```
CREATE FUNCTION ESWRAPPER.ES_SEARCH(INTEGER, VARCHAR(1024))
  RETURNS INTEGER AS TEMPLATE
```

Related reference:

- "Wrapper Extended Search - Consultas de Exemplo" na página 323
- "Wrapper Extended Search - Linguagem de Consulta Generalizada" na página 326

- “Sintaxe da Instrução CREATE FUNCTION - Wrapper Extended Search” na página 357
- “Wrapper Extended Search - Diretrizes de Consultas” na página 321

Wrapper Extended Search - Diretrizes de Consultas

O wrapper Extended Search espera que as consultas estejam em um formato específico e não suporta consultas que não atendem critérios de idiomas precisos. Este tópico fornece diretrizes para criar consultas e fornece exemplos de sintaxes de consultas corretas e incorretas.

Consultando origens da Web em vários idiomas

O software de terceiros que o Extended Search utiliza para efetuar o link com origens da Web suporta idiomas que utilizam a página de código ISO-8859-1 (como inglês, francês, alemão, português e sueco). Portanto, ao pesquisar origens da Web, você não pode pesquisar idiomas do conjunto de caracteres de byte duplo, tais como, coreano, idiomas bidirecionais, tais como, hebraico, ou outros idiomas não-ISO-8859-1. O analisador que processa resultados da pesquisa falha quando detecta o que considera como códigos de caracteres inválidos.

Especificando o valor CLIENT_LOCALE

Se você incluir a coluna CLIENT_LOCALE em uma cláusula WHERE para definir o valor do locale do cliente, será necessário utilizar um predicado AND para especificar os critérios de pesquisa. Você não pode utilizar um predicado OR com a coluna CLIENT_LOCALE.

Exemplos — sintaxe correta

Os exemplos a seguir mostram a forma correta de incluir a coluna CLIENT_LOCALE em uma cláusula WHERE:

```
WHERE CLIENT_LOCALE = 'enUS' AND
ESWRAPPER.ES_SEARCH(DOC_RANK, '"IBM"')=1
```

```
WHERE ESWRAPPER.ES_SEARCH(DOC_RANK, '"IBM"')=1
AND CLIENT_LOCALE = 'enUS'
```

Exemplos — sintaxe incorreta

Os exemplos a seguir estão incorretos porque eles tentam utilizar um predicado OR com a coluna CLIENT_LOCALE:

```
WHERE CLIENT_LOCALE = 'enUS' OR  
ESWRAPPER.ES_SEARCH(DOC_RANK, '"IBM"')=1
```

```
WHERE ESWRAPPER.ES_SEARCH(DOC_RANK, '"IBM"')=1  
OR CLIENT_LOCALE = 'enUS'
```

Especificando predicados em colunas fixas do Extended Search

Uma instrução SQL que contém um pseudônimo do Extended Search deve especificar um predicado para o pseudônimo na cláusula WHERE. No entanto, um predicado em uma coluna fixa do Extended Search não é considerado um predicado.

Exemplos — sintaxe incorreta

O exemplo a seguir mostra uma consulta que está incorreta, porque não contém um predicado:

```
SELECT * FROM ES_NICKNAME
```

O exemplo a seguir mostra uma consulta que está incorreta porque o único predicado está em uma coluna fixa:

```
SELECT * FROM ES_NICKNAME WHERE DOC_RANK < 20
```

Especificando predicados não ligados

Um predicado em uma coluna definida pelo usuário apenas será tratado pelo wrapper Extended Search se o valor do predicado for uma constante. Se o valor do predicado não estiver ligado, o predicado será tratado pelo mecanismo do DB2. Se um predicado não ligado for o único predicado em uma instrução SQL, ocorrerá um erro. Um pseudônimo do Extended Search requer um predicado que possa ser tratado pelo wrapper Extended Search.

Exemplos — sintaxe correta

A instrução WHERE no exemplo a seguir mostra um predicado que será tratado pelo wrapper Extended Search:

```
SELECT *  
FROM ES_NICKNAME  
WHERE Author = 'Ernest Hemingway'
```

Exemplos — sintaxe incorreta

A instrução WHERE no exemplo a seguir mostra um predicado que será tratado pelo DB2:

```
SELECT *  
FROM ES_NICKNAME_1, ES_NICKNAME_2  
WHERE ES_NICKNAME_1.Author = ES_NICKNAME_2.Author
```

Unindo consultas com um predicado OR

O wrapper Extended Search não pode pesquisar diferentes tabelas de pseudônimos, ou tabelas de pseudônimos e tabelas do banco de dados que são unidas por um predicado OR simples. Você pode utilizar um predicado OR apenas no mesmo pseudônimo.

Exemplos — sintaxe incorreta

```
SELECT *
FROM   ES_Nickname as N1, TABLE as T1
WHERE  N1.Column1 = 'abc' OR T1.Column1 = 'abc'
```

```
SELECT *
FROM   ES_Nickname_1 as N1, ES_Nickname_2 as N2
WHERE  N1.USerdefCol = 'abc' OR N2.USerdefCol = 'cdf'
```

```
SELECT *
FROM   ES_Nickname_1 as N1, ES_Nickname_2 as N2
WHERE  ESWRAPPER.ES_SEARCH(N1.DOC_RANK, "IBM")=1 OR
       ESWRAPPER.ES_SEARCH(N2.DOC_RANK, "LOTUS")=1
```

Related tasks:

- “Registrando a Função Personalizada do Extended Search” na página 320

Related reference:

- “Wrapper Extended Search - Consultas de Exemplo” na página 323
- “Wrapper Extended Search - Linguagem de Consulta Generalizada” na página 326

Wrapper Extended Search - Consultas de Exemplo

Para executar consultas com o wrapper Extended Search, especifique um pseudônimo registrado e as colunas de pseudônimos em suas instruções SQL da mesma forma que especificaria um nome de tabela típico do DB2 e colunas de tabela.

Neste cenário de pesquisa de amostra, uma equipe do hospital precisa pesquisar e comparar a pesquisa médica mais recente. Para pesquisar uma ampla variedade de origens, o hospital utiliza um servidor Extended Search. O domínio do Extended Search inclui um aplicativo denominado MedResearch e várias categorias que estão configuradas para pesquisar bancos de dados baseados em documentos, servidores de correio e a Web.

Além da pesquisa, a equipe precisa comparar os resultados de várias pesquisas. Por exemplo, precisam identificar pessoas que publicaram artigos em um determinado período de tempo, remédios e vitaminas comprados

recentemente, medicina alternativa discutida com colegas por e-mail e aplicada para renovar uma licença médica. O wrapper Extended Search, com sua capacidade de integrar dados não estruturados do Extended Search no DB2 para recuperação estruturada, fornece a solução.

A equipe do hospital decide criar os três pseudônimos a seguir, um para pesquisar repositórios de documentos, um para pesquisar sistemas de e-mail e um para pesquisar origens da Web específicas. Os campos Proprietário e Data são definidos como campos mapeados no banco de dados de configuração do Extended Search, que permite utilizá-los em uniões, independentemente de como os campos são denominados nas origens de dados nativas.

Pseudônimo do documento:

```
CREATE NICKNAME MedDocs ( Owner      VARCHAR(80),
                          Date       DATE,
                          Title      VARCHAR(80),
                          Abstract   VARCHAR(200) )
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                              CATEGORY 'AMA Library;Medical Records;Pharmacy',
                              VERTICAL_TABLE 'YES',
                              TIMEOUT '60', MAXHITS '100',
                              TOTALMAXHITS '1000' )
```

Pseudônimo do e-mail:

```
CREATE NICKNAME MedMail ( Owner      VARCHAR(80),
                           To        VARCHAR(80),
                           Date      DATE,
                           Subject   VARCHAR(80) )
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                              CATEGORY 'Exchange Server;Lotus Notes',
                              VERTICAL_TABLE 'YES', )
                              TIMEOUT '60', MAXHITS '100',
                              TOTALMAXHITS '1000' )
```

Pseudônimo da Web:

```
CREATE NICKNAME MedWeb ( WebTitle VARCHAR(255),
                          WebDescription VARCHAR(1000) )
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                              DATASOURCES 'Google!;Alta Vista;CNN',
                              TOTALMAXHITS '500' )
```

A consulta a seguir pesquisa documentos que contêm a frase Artificial Liver no título e a abreviação MARS no conteúdo do documento. O conjunto de resultados deve excluir documentos que foram publicados antes do ano 2001.

```
SELECT OWNER, DOC_CONTENT
FROM   MedDocs
WHERE  ESWRAPPER.ES_Search(DOC_RANK, '( ( TOKEN:EXACT "MARS") AND
                                     ( ("TITLE" IN "Artificial Liver") AND
                                       ("DATE" >= "01/01/2001") ) ) ') = 1
```

A consulta a seguir pesquisa e-mail que foi escrito durante os últimos meses que discutiam a medicina alternativa:

```
SELECT *
FROM MedMail
WHERE ESWRAPPER.ES_Search(DOC_RANK, '(
    ("SUBJECT" IN "alternative medicine") AND
    ("DATE" BETWEEN "03/01/2002" AND
    "09/30/2002") ) ') = 1
```

A consulta a seguir pesquisa origens da Web que se referem à terapia CAM (complementary and alternative medicine, medicina alternativa e complementar) e sua aceitação pelo público americano:

```
SELECT WebTitle, WebDescription
FROM MedWeb
WHERE ESWRAPPER.ES_Search(DOC_RANK, '(
    TOKEN:EXACT "CAM therapy" ) AND
    ( TOKEN:FUZZY "United States" ) ' ) = 1
```

A consulta a seguir pesquisa os médicos licenciados recentemente que compraram grandes quantidades de remédios ou vitaminas da farmácia do hospital. A consulta corresponde os nomes destes médicos com pessoas que escreveram e-mail sobre medicina alternativa.

```
SELECT N2.OWNER, N2.DATE
FROM MedDocs as N1,
MedMail as N2
WHERE ESWRAPPER.ES_SEARCH(N1.DOC_RANK, '(
    ("LICENSE_DATE" >= "01/01/2002") AND
    ( ( "PRODUCT" = "HERB") OR ("PRODUCT" = "VITAMIN") ) ) AND
    ("QUANTITY" > "1000") ) ) = 1
AND ESWRAPPER.ES_SEARCH(N2.DOC_RANK, '( "SUBJECT" IN
    "alternative medicine" ) ') = 1
AND N1.OWNER = N2.OWNER
```

Related concepts:

- “Pseudônimos do Extended Search” na página 310
- “Tabelas Verticais do Extended Search” na página 312

Related tasks:

- “Registrando Pseudônimos para Origens de Dados Extended Search” na página 317
- “Registrando a Função Personalizada do Extended Search” na página 320

Related reference:

- “Wrapper Extended Search - Linguagem de Consulta Generalizada” na página 326
- “Sintaxe da Instrução CREATE FUNCTION - Wrapper Extended Search” na página 357

- “Sintaxe da Instrução CREATE NICKNAME - Wrapper Extended Search” na página 366
- “Wrapper Extended Search - Diretrizes de Consultas” na página 321

Wrapper Extended Search - Linguagem de Consulta Generalizada

As consultas transmitidas para um servidor Extended Search através do wrapper Extended Search podem conter expressões de pesquisa em GQL (generalized query language) a linguagem de consulta do Extended Search.

Por exemplo, suponha que um usuário deseja localizar todos os funcionários cujos nomes começam com JO em um banco de dados relacional que contenha uma tabela com informações sobre funcionários. Você pode emitir a seguinte consulta no GQL:

```
(LIKE "EMPLOYEE_NAME" "JO")
```

Você pode emitir a mesma consulta no SQL da seguinte forma:

```
SELECT * FROM EMP.TABLE WHERE EMPLOYEE_NAME LIKE JO%
```

Assim com o SQL, o wrapper suporta notação de afixo, uma sintaxe que requer que os operadores estejam entre o nome do campo e um valor de comparação. A gramática GQL nativa do Extended Search utiliza notação de prefixo, uma sintaxe que requer que os operadores precedam os campos e valores que você deseja avaliar.

Compare as seguintes expressões de consulta que pesquisam documentos que contêm a palavra IBM no campo TITLE:

Infix GQL

```
("TITLE" IN "IBM")
```

Prefixo GQL

```
(IN "TITLE" "IBM")
```

Ao enviar uma consulta com o wrapper Extended Search, a API converte as instruções SQL de afixo em GQL de prefixo para processamento pelo Extended Search.

A descrição de sintaxe a seguir mostra a especificação Backus-Naur Form para a gramática do Extended Search que pode ser utilizada em consultas.

```
expr:          pattern_expr
              | bool_expr
              | field_expr
              | prox_expr
```

```

pattern_expr:  STRING
              | token_expr

token_expr:   ( TOKEN [:CASE] [:STEM] [:EXACT] [:WEIGHT "x"]
              [:WILD] [:FUZZY] STRING )

bool_expr:    (expr_list bool_operator [:WEIGHT "x"] expr )

bool_text_expr: (text_expr_list bool_operator [:WEIGHT "x"] text_expr )

text_expr:    pattern_expr
              | bool_text_expr
              | prox_expr

text_expr_list: text_expr
                | text_expr_list text_expr

expr_list:    expr
              | expr_list expr

field_expr:   ( field_name operator_1 [:WEIGHT "x"] text_expr )
              | ( field_name operator_2 [:WEIGHT "x"] value )
              | ( field_name operator_3 [:WEIGHT "x"] value_1 AND value_2 )
              | ( field_name operator_4 value )

prox_expr:    ( prox_op [:COUNT "x"][:ORDER][:MATH "y"][:WEIGHT "x"]
              expr_list expr )

prox_op:      DOCUMENT
              | PARAGRAPH
              | SENTENCE
              | WORD
              | CHARACTER

operator1:    START
              | END
              | IN
              | =

operator_2:   =
              | >
              | >=
              | <
              | <=
              | EQ
              | GT
              | GTE
              | LT
              | LTE

operator_3:   BETWEENI
              | BETWEENE
              | LIKE

```

```
bool_operator:  AND
                |  OR
                |  NOT
```

Para obter informações completas sobre a gramática GQL, consulte *Extended Search Programming*, que está disponível na página Resources do Web site IBM Lotus Extended Search:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

Related tasks:

- “Registrando a Função Personalizada do Extended Search” na página 320

Related reference:

- “Wrapper Extended Search - Consultas de Exemplo” na página 323
- “Sintaxe da Instrução CREATE FUNCTION - Wrapper Extended Search” na página 357
- “Wrapper Extended Search - Diretrizes de Consultas” na página 321

Mensagens para o Wrapper Extended Search

Este tópico descreve mensagens que podem ser encontradas ao trabalhar com o wrapper Extended Search.

Tabela 59. Mensagens emitidas pelo wrapper para Extended Search

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL0901N | A instrução SQL falhou devido a um erro de pouca gravidade do sistema. As instruções SQL posteriores podem ser processadas. (Razão: ERRO INTERNO DO WRAPPER Extended Search - RC: xxx.) | Registre o código de razão (um número de 901 a 999) e entre em contato com o Suporte de Software IBM. |

Tabela 59. Mensagens emitidas pelo wrapper para Extended Search (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL0973N | Não há armazenamento suficiente disponível no heap do Aplicativo para processar a instrução. | <p>O wrapper Extended Search não pôde alocar memória no heap do Aplicativo. Para resolver o problema, aumente o tamanho de heap do Aplicativo e tente a instrução novamente. Por exemplo:</p> <pre>db2 update db cfg for db-name using heap-name heap-size</pre> <p>Se o erro continuar depois de aumentar este valor, entre em contato com o Suporte de Software da IBM.</p> |
| SQL1822N | Código de erro inesperado "<error_code>" recebido da origem de dados "Wrapper Extended Search". O texto e tokens associados são "<tokens>". | O servidor remoto Extended Search retornou um erro ao processar um pedido de pesquisa. O erro também retornou um token que indica o que causou o erro no servidor remoto. Se o rastreo estiver ativado para o servidor Extended Search, reveja o arquivo de log de rastreo para obter ajuda para diagnóstico. |
| SQL1823N | Não existe mapeamento de tipo de dados para o tipo de dados "<tipo_de_dados>" do servidor "<nome_do_servidor>". | Uma coluna em uma instrução CREATE NICKNAME ou ALTER NICKNAME utiliza um tipo de dados não suportado pelo sistema Extended Search. Este erro também pode ocorrer durante o processamento da consulta. Para resolver o problema se ele ocorrer durante o processamento da consulta, elimine a tabela do pseudônimo e crie um novo pseudônimo. |
| SQL1825N | Essa instrução SQL não pode ser manipulada em um ambiente federado. | A instrução SQL atual não pode ser tratada pelo wrapper Extended Search. Para resolver o problema, consulte a documentação do wrapper Extended Search, altere a instrução SQL conforme necessário e envie o pedido novamente. |

Tabela 59. Mensagens emitidas pelo wrapper para Extended Search (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|---|
| SQL1833N | A conexão com o servidor remoto Extended Search " <code><nome_do_host></code> " na porta " <code><número_da_porta></code> " não pôde ser estabelecida ou foi finalizada. | O wrapper Extended Search tentou conectar-se ao servidor remoto Extended Search na porta especificada, mas a conexão não pôde ser estabelecida ou foi finalizada pelo servidor remoto. Verifique o nome do host e o número da porta do servidor remoto Extended Search, certifique-se de que o servidor Extended Search esteja em execução e tente novamente. |
| SQL1834N | A coluna definida pelo usuário " <code><nome_da_coluna></code> " é idêntica a uma coluna fixa para o wrapper " <code><wrapper_name></code> ", mas utiliza um tipo de dados diferente. | Uma instrução CREATE NICKNAME ou ALTER NICKNAME contém uma coluna definida pelo usuário que possui o mesmo nome que uma coluna fixa para o wrapper Extended Search especificado, mas utiliza um tipo de dados diferente. Você não precisa especificar colunas fixas na definição de coluna de uma instrução CREATE NICKNAME. Se especificar, certifique-se de que o nome da coluna fixa, o tipo de dados e o comprimento do tipo de dados correspondem à definição de coluna fixa. Não é possível ALTERAR um nome de coluna fixa ou um tipo de dados. |
| SQL1835N | O objeto do Extended Search " <code><object_name></code> " de tipo " <code><object_type></code> " não pôde ser localizado no servidor remoto Extended Search " <code><nome_do_host></code> ". | O objeto do Extended Search especificado não pôde ser localizado no servidor remoto Extended Search especificado. Verifique se o nome do objeto está definido neste servidor Extended Search e se ele possui o tipo de objeto especificado. Verifique também se a ortografia deste objeto está correta. |
| SQL1836N | Não existe nenhum mapeamento de coluna entre a coluna definida pelo usuário " <code><nome_da_coluna></code> " e um nome de campo no servidor remoto Extended Search " <code><nome_do_host></code> ". | Nenhuma das origens de dados incluídas em uma opção DATASOURCE ou CATEGORY contém um nome de campo correspondente ao nome da coluna definida pelo usuário especificado. Verifique se o nome da coluna é um campo em, pelo menos, uma das origens de dados na opção DATASOURCE ou em, pelo menos, uma das origens de dados pertencentes a uma categoria na opção CATEGORY e envie a instrução novamente. |

Tabela 59. Mensagens emitidas pelo wrapper para Extended Search (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|--|---|
| SQL1837N | A opção requerida " <code><option_name></code> " de tipo " <code><object_type></code> " no wrapper " <code><wrapper_name></code> " não pode ser eliminada. | Não é possível eliminar uma opção requerida. Altere a instrução ALTER para utilizar SET em vez de DROP. Corrija a instrução de pesquisa e envie novamente o pedido. Consulte o DB2 SQL Reference para obter informações sobre como criar instruções de pesquisa SQL válidas. Se a instrução de pesquisa incluir a função ES_SEARCH, consulte a documentação do wrapper Extended Search para obter informações sobre como utilizar o GQL (generalized query language) do Extended Search. |
| SQL1838N | A instrução de pesquisa " <code><option_name></code> " não é uma consulta do Extended Search válida. | O wrapper do Extended Search tentou processar a instrução de pesquisa especificada, mas a consulta falhou porque a instrução não utiliza a sintaxe de consulta correta. Consulte o <i>DB2 SQL Reference</i> para obter informações sobre como criar instruções de pesquisa SQL válidas. Se a instrução de pesquisa incluir a função ES_SEARCH, consulte a documentação do wrapper Extended Search para obter informações sobre como utilizar o GQL (generalized query language) do Extended Search. |
| SQL1839N | Um ou mais parâmetros de pesquisa não são válidos. | O wrapper de Pesquisa Estendida tentou utilizar os parâmetros de pesquisa especificados, mas eles não são válidos para Pesquisa Estendida. Consulte a documentação do wrapper Extended Search, corrija os parâmetros inválidos e envie o pedido novamente. |
| SQL1881N | " <code><nome_da_opção></code> " não é uma opção válida do " <code><tipo_de_opção></code> " para " <code><nome_do_objeto></code> ". | A opção especificada não é válida para o objeto especificado (wrapper, servidor, pseudônimo, coluna ou mapeamento de usuário). Consulte a documentação do wrapper Extended Search, remova ou altere a opção válida e envie a instrução novamente. |

Tabela 59. Mensagens emitidas pelo wrapper para Extended Search (continuação)

| Código de Erro | Mensagem | Explicação |
|----------------|---|---|
| SQL1882N | A opção "<option_type>" "<option_name>" não pode ser definida como "<option_value>" para "<object_name>". | O valor de opção especificado é inválido para o objeto especificado (wrapper, servidor, pseudônimo, coluna ou mapeamento de usuário). Consulte a documentação do wrapper Extended Search, altere o valor de opção inválido e envie a instrução novamente. |
| SQL1883N | "<nome_da_opção>" é uma opção "<tipo_de_opção>" requerida para "<nome_do_objeto>". | Uma opção requerida para o wrapper Extended Search estava ausente na instrução para criar, alterar ou inicializar o objeto especificado (wrapper, servidor, pseudônimo ou mapeamento de usuário). Consulte a documentação do wrapper Extended Search, adicione a opção requerida e envie a instrução novamente. |

Para obter mais informações sobre mensagens, consulte o *DB2 Message Reference*. Talvez você também queira consultar as mensagens do produto Extended Search no *Extended Search Administration*. Se receber erros sobre sintaxe incorreta de consulta GQL, consulte o *Extended Search Programming*. Os documentos do Extended Search estão disponíveis na página Resources do Web site do IBM Lotus Extended Search:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

Related reference:

- "sql0900" no *Message Reference: Volume 2*
- "sql1800" no *Message Reference: Volume 2*

Capítulo 19. Configurando o Acesso às Origens de Dados do HMMER

Este capítulo explica o que é o HMMER, como incluir origens de dados do HMMER em seu sistema federado e lista as mensagens de erro associadas ao wrapper do HMMER.

O que É HMMER?

HMMER é uma ferramenta que pode ser utilizada para pesquisar bancos de dados de seqüência de genes que utilizam modelos estatísticos ou HMMs (Hidden Markov Models) de perfil. Você pode fazer o download do HMMER gratuitamente a partir de <http://hmmer.wustl.edu/>. O HMMER foi desenvolvido primeiramente para aprimorar os recursos de pesquisa BLAST. Esta versão do wrapper HMMER utiliza uma seqüência de gene para pesquisar um banco de dados de modelos e determinar a qual família a seqüência de genes de teste pode pertencer.

Um HMM é um modelo estatístico do consenso de estrutura primária de uma família de seqüência de gene. Um HMM é baseado em modelos de probabilidade. Você pode treinar um HMM para reconhecer padrões de seqüências de genes não alinhados se um alinhamento confiável ainda não for conhecido. Você precisa de menos habilidade e intervenção manual para treinar e utilizar um HMM bem-sucedido do que para construir um perfil cuidadosamente. Você pode utilizar um HMM treinado para acessar bibliotecas e centenas de HMMs de perfil e aplicá-los em grande escala a análises de todo o genoma ou de EST (Expressed Sequence Tag).

PFAM (Protein Families Database of Alignments and HMMs) é um banco de dados de modelos de domínio de proteínas. O pacote de software HMMER está firmemente ligado à construção e utilização do banco de dados PFAM. O wrapper HMMER suporta a utilização do programa **hmmpfam**, que pesquisa um banco de dados HMM de perfil, como PFAM, com uma seqüência de genes específica.

O wrapper HMMER inicia o utilitário **hmmpfam**, que utiliza HMMs do perfil para modelar o consenso de estrutura primário de uma família de seqüências de ácidos protéico ou nucléico, conforme descrito na Tabela 60 na página 334.

Tabela 60. Utilitários HMMER

| Utilitário HMMER | Descrição |
|------------------|--|
| hmmpfam | Calcula como cada modelo corresponde a uma seqüência especificada e a um banco de dados de modelos. A correspondência é expressa em termos de importância estatística. |
| hmmalign | Alinha várias seqüências de genes em um HMM de perfil. |
| hmmbuild | Constrói um HMM de perfil a partir de um alinhamento de seqüência de genes. |
| hmmcalibrate | Determina os parâmetros de importância estatística apropriados para um HMM de perfil antes da execução de uma pesquisa do banco de dados. |
| hmmconvert | Converte HMMs de perfil HMMER em outros formatos, tais como, perfis GCG (Genetics Computer Group). |
| hmmemit | Gera seqüências de genes que utilizam modelos de probabilidade de um HMM de perfil. |
| hmmfetch | Recupera um HMM de um banco de dados HMM. |
| hmindex | Cria um índice de inclusão binária adicional do servidor SSI (inclusões adicionais do servidor) para um banco de dados HMM. |
| hmmsearch | Pesquisa um banco de dados de seqüência de genes com um HMM de perfil e localiza homólogos de uma família modelada. |

Em um cliente, os usuários ou aplicativos enviam instruções SQL com predicados específicos de HMMER que são mapeados para opções da linha de comandos de hmmpfam. Estas instruções SQL e predicados são enviados para o servidor do banco de dados federado, que inclui o wrapper HMMER.

O wrapper HMMER transforma a consulta em um formato que o aplicativo HMMER pode interpretar e inicia o utilitário hmmpfam para executar a consulta. O servidor que executa hmmpfam pode ser um sistema separado do sistema com o servidor do banco de dados federado. Um programa de daemon especial é executado no servidor HMMER. Este daemon, que utiliza informações de um arquivo de configuração do daemon, recebe o pedido de consulta do servidor do banco de dados federado e envia-o para o aplicativo HMMER. Em seguida, o aplicativo HMMER é executado em um banco de dados de perfil.

A Figura 14 mostra como o HMMER funciona com o sistema federado.

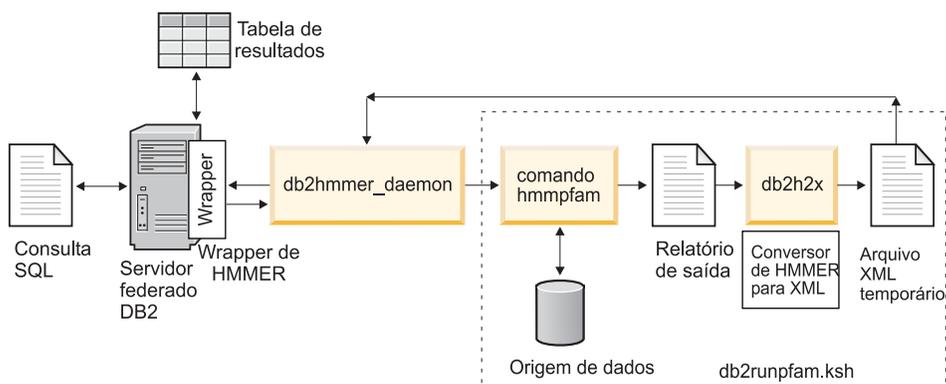


Figura 14. Como funciona o wrapper HMMER

O daemon retorna resultados do HMMER para o wrapper HMMER. O wrapper transforma os dados em uma tabela relacional e retorna esta tabela ao usuário ou aplicativo.

O exemplo a seguir mostra como as informações são extraídas de bancos de dados de perfil, que são construídos por utilitários HMMER e exibidos como uma tabela relacional. O HMMER User's Guide (<http://hmmmer.wustl.edu/>) fornece exemplos de como criar bancos de dados de perfil e um tutorial do HMMER.

A Figura 15 na página 336 mostra uma consulta de amostra que utiliza a seqüência de genes 7LES_DROME. Você especifica seqüências na cláusula WHERE da consulta.

```

SELECT Model, ModelScore, DomainNumber, DomainScore
FROM myhmms
WHERE HmmQSeq = 'MTMFWQQNVHDHQSDEQDKQAKGAAPT KRLNISFNVKI AVNVNTKMTTTH
INQQAPGTSSSSSNQNASPSKI VVRQQSSFDLRQQLARLGRQLASGQDGHGGISTILI INLLLL
ILLSICDCVCRSHNYTVHQSPPEVSKDQMRLLRPKLDSDVVEKVAIWHKHA AAPPSIVEGIAISS
RPQSTMAHHPDDRDRDRDPSEEQHGVDERMVLERVTRDCVQRCIVEEDLFLDFEFGIQCEKADNGEK
CYKTRCTKGCQAQWYRALKELESCQEACLSLQFYYPDMPCIGACEMAQRDYWHLQRLAISHLVERTQ
PQLERAPRADGGSTPLTIRWAMHFPEHYLASRPFNIQYQFVDHGHGEELEDEQEDQDASGETGSSAW
FNLADYDCDEYMYCEILEALIPYTQYRFRFELPFGENRDEVLYSPATPAYQTPPEGAPISAPVIEH
LMGLDDSHLAVHWHGPRFTNGPIEGYRLRLSSSEGNATSEQLVPAGRGSYIFSQLQAGTNYTLALS
MINKQGEGPVAKGQVQTHSARNEKPAKDLTESVLLVGRRAVMWQSLPAGENSMIYQSQEELADIA
WSKREQLLWLLNVHGLRSLKFESGQMVSPAQQLKLDLGNISSGRWVPRRSLFDWLHHRLYFAMES
PERNQSSFOIISTDILGESAQKVGESFDLPVEQLEVDALNGWIFWRNEESLWRQDLHGRMIHRLLR
IRQPGWFLVQPQHFIIHMLPQEGKFLIISYDGGFKHPLPLPPPSNGAGNGPASSHWQSFALLGRS
LLL PDSGQLILVEQQQAASPSASWPLKNLPDCWAVILLVPESQPLTSAGGKPHSLKALLGAQAAK
ISWKEPERNPYQSADAARSWSYELEVLVDVASQSASFIRNIRGPIFGLQRLQPDNLYQLRVRAINVD
GEPGEWTEPLAARTWPLGPHRLRWASRQGSVIHTNELGEGLEVQEQLERLPGPMTMVNESVGYVY
TGDGLLHCINLVHSQWGCPISEPLQHVGSVTYDRWGRVYVWTDLARNCVVRMDPWSGSELLPVFE
ANFLALDPRQGHLYYATSSQLSRHGSTPDEAVTYRVNGLEGSIASFVLDTQQDQLFWLVKGS GAL
RLYRAPLTAGGDSLQMIQQIKGVFQAVPDSLQLLRPLGALLWLESGRRARLVRLAAPLDMELPT
PDQASPASALQLLDPQLPPRDEGVI PMTVLPDSVRLDDGHWDHFVWRWQPSTSGGNHSVSYRLL
EFGQLRQLTLDLSTPFARLTQLPQAQLKISITPRTAWRSGDTRRVTTPVAPSPRRRLRVFVE
RLATALQEANVSAVLRWDAPEQQQEAPMQALEYHISCWVGSSELHEELRLNQSALEARVEHLQPDQT
YHFQVEARVAATGAAAGAASHALHVAPEVQAVPRVLYANA EFIGELDLDRNRRLVHTASVPEHL
VGIEGEQRLWVNEHVELLTHVPGSAPAKLARMRAEVLALAVDWIQRIVYWAELDAPQAAIYR
LDLCNFEGLIKQGERVWSTPRGRLLKDLVALPQAQSLIWLVEYEQGSPRNGSLRGRNLDGSELEWA
TVQPLIRLHAGSLEPGSETLNLVDNQGKLCVYDVARQLCTASALRAQLNLLGEDS TAGQLAQDSGY
LYAVKNWSIRAYGRRRQQLLEYTVELEPEEVRLLQAHNYQAYPPKNCLLLPSSGGSLLKATDCEEQR
CLLNLPMITASEDCPLPIPGVRYQLNLTARGPGSEEHDHGVEPLGQWLLGAGESLNLTDLLPFTR
YRVSGILSSFYQKALPTLVLAPELLETTASATPSPPRNFVSRVLSPRELEVSWLPPEQLRSESVY
YTLHWQQLDGENVQDRREWEAHERRLETAGTHRLTGIKPGSGYSLWVQAHATPTKSNSSERLHVR
SFAELPELQLELGPYSLSLTWAGTPDPLGSLQLECRSSAEQLRRNVAGNHTKMVVEPLQPRTRYQ
CRLLLGYAATPGAPLYHGTAEVYETLGDAPSQPGKPKOLEHIAAEVFRVITWTAAARGNGAPITALYNE
ALQARSDIRRRRRRRRRNSGGSLQLPWAEPPVVVEDQWLDFCNTTELSCIVKSLHSSRLLLFRVR
ARSLHGWGPYSEESERVAEPFVSPEKRGSLVLAIIAPAAIVSSCVLALVLRVKVQKRRLRRAKLL
QQSRPISWNSLSTLQTQQQLMAVRNRAFSTTSLSDADIALLPQINWSQLKLLRFLGSGAFGEVYEGQ
LKTEDSEEPQRAIKSLRKGASEFAELLQEAQLMSNFKHENIVRLVIGICFDTESISLIMEHMEAGD
LLSYLRAARATSTQEPQTAGLSSELLAMCIDVANGCSYLEDMHFVHRDLACRNLVTESTGSTD
RRRTVKIGDFGLARDIYKSDYRKEGEGLLPVRWMSPESLVDGLFTTQSDVWAFVGLCWEILTLGQ
QPYAARNNEFVLAHVKEGGRLQPPMCTEKLYSLLLLCWRTDPWERPSFRRCYNTLHAISTDLRRT
QMASATADTVVSCSRPEFKVRFDGQPLEEHREHNERPEDENLTLREVPLKDKQLYANEGVSRL'
```

Figura 15. Consulta de amostra nos dados 7LES_DROME

O wrapper HMMER transforma os resultados da consulta na tabela relacional conforme mostrado na Tabela 61.

Tabela 61. O HMMER retorna resultados em uma tabela relacional quando o wrapper HMMER está integrado ao sistema federado

| Modelo | ModelScore | DomainNumber | DomainScore |
|---------|------------------------|--------------|------------------------|
| pkinase | +3.04100000000000E+002 | 1 | +3.04100000000000E+002 |

Tabela 61. O HMMER retorna resultados em uma tabela relacional quando o wrapper HMMER está integrado ao sistema federado (continuação)

| Modelo | ModelScore | DomainNumber | DomainScore |
|--------|-------------------------|--------------|-------------------------|
| fn3 | +1.763000000000000E+002 | 1 | +4.900000000000000E+001 |
| fn3 | +1.763000000000000E+002 | 2 | +1.360000000000000E+001 |
| fn3 | +1.763000000000000E+002 | 3 | +1.620000000000000E+001 |
| fn3 | +1.763000000000000E+002 | 4 | +6.350000000000000E+001 |
| fn3 | +1.763000000000000E+002 | 5 | +1.460000000000000E+001 |
| fn3 | +1.763000000000000E+002 | 6 | +1.940000000000000E+001 |
| rrm | -4.450000000000000E+001 | 1 | -4.450000000000000E+001 |

Os dados agora estão em um formulário totalmente relacional e podem ser unidos com dados de outras origens de dados.

Related concepts:

- “O que São Arquivos com Estrutura de Tabela?” na página 151
- “O que É Documentum?” na página 165
- “O que É Excel?” na página 203
- “O que É BLAST?” na página 219
- “O que É XML?” na página 247
- “O que É Entrez?” na página 279
- “O que É Extended Search?” na página 307

Adicionando HMMER a um Sistema Federado

Procedimento:

Para adicionar a origem de dados HMMER a um servidor federado:

1. Verifique se foi instalada a versão correta do executável hmmpfam.
2. Configure o daemon HMMER. Configure o daemon HMMER.
3. Inicie o daemon HMMER. Inicie o daemon HMMER.
4. Registre o wrapper, emitindo a instrução CREATE WRAPPER. Registre o wrapper, emitindo a instrução CREATE WRAPPER.
5. Opcional: Defina a variável de ambiente DB2_DJ_COMM para melhorar o desempenho da consulta.
6. Registre o servidor, emitindo a instrução CREATE SERVER. Registre o servidor, emitindo a instrução CREATE SERVER.

7. Registre pseudônimos, emitindo a instrução CREATE NICKNAME. Registre pseudônimos, emitindo a instrução CREATE NICKNAME.

Você pode executar instruções a partir do processador da linha de comandos do DB2. Depois de adicionar o wrapper HMMER a seu sistema federado, você pode executar consultas na origem de dados HMMER.

Related tasks:

- “Verificando se a Versão Correta do Executável hmmpfam Está Instalada” na página 338

Verificando se a Versão Correta do Executável hmmpfam Está Instalada

Verificar se a versão correta do executável hmmpfam está instalada faz parte da maior tarefa de adicionar HMMER a um sistema federado. Verifique se você possui a versão mais recente do executável hmmpfam instalada no servidor HMMER com o seguinte procedimento.

Procedimento:

Para verificar o nível de versão do executável hmmpfam:

1. Execute o seguinte a partir da linha de comandos e anote o número da versão localizado no arquivo de saída:

```
hmmpfam -h
```
2. Se não tiver a versão mais recente do executável hmmpfam (HMMER 2.2 ou superior), faça download dos arquivos a partir de <http://hmmer.wustl.edu/>.

A próxima tarefa nesta seqüência de tarefas é configurar o daemon HMMER.

Related tasks:

- “Configurando o Daemon HMMER” na página 338

Configurando o Daemon HMMER

A configuração do daemon HMMER faz parte da maior tarefa de adicionar HMMER a um sistema federado. O wrapper HMMER requer que um daemon HMMER seja executado na máquina baseada em AIX. Você também deve poder acessar o daemon HMMER através do TCP/IP (Transmission Control Protocol/Internet Protocol) a partir do servidor federado. O daemon é executado separadamente do wrapper e do DB2 Universal Database e atende pedidos do job HMMER a partir do wrapper. O arquivo executável do daemon, `db2hmmer_daemon`, pode residir em qualquer diretório no servidor HMMER.

Durante a instalação do DB2 Universal Database, o executável do daemon é colocado no mesmo computador que o servidor federado. Na plataforma AIX, o diretório no qual o executável do daemon é colocado é `/usr/opt/db2_08_01/bin`. Se você não instalou o HMMER e o servidor federado no mesmo computador, deverá copiar o executável do daemon para uma localização de sua escolha no computador no qual o HMMER foi instalado. Você também deve copiar o arquivo de configuração `HMMER_DAEMON.config`, o utilitário de conversão fornecido denominado `db2h2x` e o script de shell denominado `db2runpfam.ksh`.

Os programas devem ser executáveis. Se não forem, execute o comando do AIX `chmod a+x db2hmmmer_daemon db2h2x db2runpfam.ksh` no sistema de destino para tornar os programas executáveis.

O daemon HMMER deve ter:

- Acesso de execução ao arquivo executável `hmmmpfam` para que ele possa executar pesquisas do HMMER.
- Acesso de gravação a um diretório no qual ele possa gravar arquivos temporários.
- Acesso de leitura a, pelo menos, um banco de dados do perfil no qual você possa executar pesquisas de HMMER.

O daemon HMMER requer um arquivo de configuração. Um arquivo de configuração do daemon de amostra, denominado `HMMER_DAEMON.config`, é colocado no diretório `DB2PATH/samples/lifesci`, em que `DB2PATH` é o diretório no qual o DB2 Universal Database está instalado. `HMMER_DAEMON.config` é o nome padrão do arquivo.

Copie o arquivo de configuração para qualquer localização acessível ao daemon, renomeie-o, se desejar e edite-o para funcionar com suas origens de dados. Por padrão, o daemon HMMER procura suas informações de configuração no diretório de trabalho a partir do qual foi iniciado.

Procedimento:

Para configurar o daemon, especifique as seguintes opções no arquivo de configuração. Para opções que requerem caminhos, você pode especificar caminhos relativos. Caminhos relativos são relativos ao diretório a partir do qual o processo do daemon foi iniciado.

DAEMON_PORT

Esta é a porta da rede na qual o daemon atenderá pedidos do job HMMER enviados pelo wrapper.

MAX_PENDING_REQUESTS

Este é o número máximo de pedidos do job HMMER que podem ser

bloqueados no daemon a qualquer momento. Este número não representa o número de jobs HMMER que são executados ao mesmo tempo, somente o número de pedidos do job que podem ser bloqueados de uma só vez. Recomendamos que você defina este valor para um número maior que cinco. O daemon HMMER não restringe o número de jobs HMMER que podem ser executados ao mesmo tempo.

DAEMON_LOGFILE_DIR

Este é o diretório no qual o daemon criará seu arquivo de log. Este arquivo conterá informações úteis de status e de erros geradas pelo daemon HMMER.

Q_SEQ_DIR_PATH

Este é o diretório no qual um arquivo temporário de dados de seqüência de consulta será criado pelo daemon. Este arquivo temporário é limpo quando o job HMMER é concluído.

HMMER_OUT_DIR_PATH

Este é o diretório no qual o daemon criará o arquivo temporário para armazenar os dados de saída de HMMER. Os dados serão lidos a partir deste arquivo e passados de volta para o wrapper através da conexão de rede, neste ponto o daemon elimina o arquivo temporário.

RUNPFAM_PATH

Este é o nome completo do script de shell db2runpfam.ksh fornecido.

HMMPFAM_PATH

Este é o nome completo do arquivo executável hmmpfam na máquina que executa o daemon.

H2X_PATH

Este é o nome completo do programa de conversão db2h2x (HMMER em XML) fornecido com o daemon.

entrada de especificação do banco de dados

Especifica a localização de um banco de dados do perfil. Ao criar um pseudônimo para a origem dados com a instrução CREATE NICKNAME, anote o nome da entrada utilizado na opção DATASOURCE do arquivo de configuração. Você deve especificar este nome de entrada para o daemon funcionar corretamente.

O arquivo de configuração deve conter, pelo menos, uma entrada de especificação de banco de dados no seguinte formato:

```
entry_name = path to profile_database
```

Por exemplo, para especificar o banco de dados do perfil MYHMMS, adicione a seguinte linha no arquivo de configuração do daemon:

```
myhmms=/home/user_ID/myhmms
```

O arquivo de configuração deve terminar com um caractere de nova linha.

Exemplo:

O exemplo a seguir mostra o conteúdo de um arquivo de configuração de amostra, com as opções requeridas e a especificação do banco de dados do perfil para PFAM.

```
=  
DAEMON_PORT=4098  
MAX_PENDING_REQUESTS=10  
DAEMON_LOGFILE_DIR=./  
Q_SEQ_DIR_PATH=./  
HMMER_OUT_DIR_PATH=./  
RUNPFAM_PATH=./db2runpfam.ksh  
HMMPFAM_PATH=/home/user_id/hmmer/bin/hmmpfam  
H2X_PATH=/home/user_id/sql1lib/bin/db2h2x  
myhmms=/home/user_id/hmmer/tutorial/myhmms  
pfam1s=/home/user_id/hmmer/pfam/Pfam_1s
```

1. Certifique-se de iniciar a primeira linha com um sinal de igual ou o daemon não será iniciado. Você obterá uma mensagem de erro, a menos que especifique DAEMON_PORT.
2. Certifique-se de terminar a última linha no arquivo de configuração com uma nova linha. Caso contrário, você obterá uma mensagem de erro quando executar uma consulta HMMER que utiliza a origem de dados listada na última linha.

A próxima tarefa nesta sequência é iniciar o daemon HMMER.

Related tasks:

- “Iniciando o Daemon HMMER” na página 341

Iniciando o Daemon HMMER

O início do daemon HMMER faz parte da maior tarefa de adicionar HMMER a um sistema federado. Antes de acessar origens de dados HMMER, é necessário iniciar o daemon HMMER.

Pré-requisitos:

Antes de iniciar o daemon HMMER, é necessário ter acesso de gravação a todos os caminhos listados nas entradas DAEMON_LOGFILE_DIR, HMMER_OUT_DIR_PATH e Q_SEQ_DIR_PATH no arquivo de configuração.

Procedimento:

Se as seguintes condições forem verdadeiras:

- Você está no diretório de instalação do daemon.
- Você não alterou o nome do arquivo de configuração do daemon.

- Colocou o arquivo de configuração no mesmo diretório que o arquivo executável do daemon.

Digite o seguinte na linha de comandos para iniciar o daemon HMMER:

```
db2hmmmer_daemon
```

O arquivo executável inicia um novo processo no qual o daemon HMMER é executado.

Se você alterou o nome do arquivo de configuração ou a localização do diretório, utilize a opção `-c` no comando do daemon do wrapper para apontar o executável do daemon para o novo nome ou localização.

Por exemplo, o comando a seguir faz com que o daemon do wrapper procure suas informações de configuração em um arquivo chamado `HMMER_D.config` no subdiretório `cfg`.

```
db2hmmmer_daemon -c cfg/HMMER_D.config
```

A próxima tarefa nesta seqüência de tarefas é registrar o wrapper HMMER.

Related tasks:

- “Registrando o Wrapper HMMER” na página 342

Registrando o Wrapper HMMER

O registro do wrapper HMMER faz parte da maior tarefa de adicionar HMMER a um sistema federado. Você deve registrar o wrapper para acessar uma origem de dados. Wrappers são mecanismos que os servidores federados utilizam para se comunicar com as origens de dados e recuperar dados delas. Seu computador instala wrappers como arquivos de biblioteca.

Procedimento:

Para registrar o wrapper HMMER, emita a instrução `CREATE WRAPPER`.

Por exemplo, para criar um wrapper HMMER no AIX chamado `my_hmmmer`, a partir do arquivo de biblioteca padrão, `libdb21shmmmer.a`, envie a seguinte instrução:

```
CREATE WRAPPER my_hmmmer LIBRARY 'libdb21shmmmer.a'  
  OPTIONS(DB2_FENCED 'N');
```

Para Windows, utilize `db21shmmmer.dll` em vez de `libdb21shmmmer.a`.

A próxima tarefa nesta seqüência de tarefas é definir a variável de ambiente `DB2_DJ_COMM` para o wrapper HMMER.

Related tasks:

- “Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper HMMER” na página 343

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper HMMER

A definição da variável de perfil DB2_DJ_COMM do DB2 para o wrapper HMMER é uma tarefa opcional da maior tarefa de adicionar HMMER a um sistema federado. Para aprimorar o desempenho ao acessar origens de dados HMMER, defina a variável de perfil DB2_DJ_COMM do DB2. Esta variável determina se o servidor federado carregará o wrapper na inicialização.

Procedimento:

Para definir a variável de perfil DB2_DJ_COMM do DB2, envie o comando `db2set` com a biblioteca do wrapper especificada na instrução `CREATE WRAPPER` associada.

Por exemplo:

```
db2set DB2_DJ_COMM=libdb21shmmmer.a,libdb21shmmmerF.a,libdb21shmmmerU.a
```

Em que `libdb21shmmmer.a`, `libdb21shmmmerF.a` e `libdb21shmmmerU.a` se referem a nomes de bibliotecas principais, limitadas e ilimitadas em uma plataforma AIX.

Verifique se não existem espaços em nenhum lado do sinal de igual (=).

Para evitar sobrecarga desnecessária ao carregar bibliotecas do wrapper na inicialização do banco de dados, especifique apenas as bibliotecas que deseja acessar.

A próxima tarefa nesta seqüência de tarefas é registrar o servidor para uma origem de dados HMMER.

Related tasks:

- “Registrando o Servidor para uma Origem de Dados HMMER” na página 344

Registrando o Servidor para uma Origem de Dados HMMER

O registro do servidor para uma origem de dados HMMER faz parte da maior tarefa de adicionar HMMER a um sistema federado. Depois de registrar o wrapper, você deve registrar um servidor correspondente.

Procedimento:

Para registrar o servidor HMMER no sistema federado, utilize a instrução CREATE SERVER.

Você deve registrar cada servidor no qual é executada uma pesquisa HMMER que utiliza o executável hmmpfam e a instância do daemon.

Por exemplo, para um wrapper chamado my_hmmer que utiliza a instrução CREATE WRAPPER para pesquisas hmmpfam, registre o servidor hmmer_server1 com a seguinte instrução:

```
CREATE SERVER hmmer_server1
  TYPE pfam
  VERSION 2.2
  WRAPPER my_hmmer
  OPTIONS (NODE 'someserver.someschool.edu', DAEMON_PORT '4098')
```

Argumentos

TYPE Obrigatório: Determina o tipo de pesquisa executada que utiliza o servidor especificado. Neste release, o valor deve ser definido como PFAM.

VERSION

Obrigatório: Especifica a versão do servidor, que deve corresponder à versão do executável hmmpfam que está em execução (HMMER 2.2 ou superior).

WRAPPER

Obrigatório: Especifica o nome do wrapper registrado durante a emissão da instrução CREATE WRAPPER.

Opções

Você deve colocar os valores de opções de servidores entre aspas simples.

NODE

Obrigatório: Especifica o nome do host do servidor no qual o processo do daemon HMMER é executado.

DAEMON_PORT

Opcional: Especifica o número da porta na qual o daemon atende pedidos de job HMMER. O número da porta deve ser igual ao

número especificado na opção DAEMON_PORT do arquivo de configuração do daemon. O padrão é 4098.

PROCESSORS

Opcional: Especifica o número de processadores utilizados pelo programa HMMER. Esta opção é equivalente à opção `--cpu` do comando `hmmpfam`. Exemplo: `PROCESSORS '2'`.

HMMPFAM_OPTIONS

Opcional: Especifica opções de `hmmpfam`, tais como, `--null2`, `--pvm` e `--xnu` que não possuem nome de coluna correspondente em uma tabela de referência que mapeia opções para nomes de colunas. Exemplo: `HMMPFAM_OPTIONS '--xnu --pvm'`. Neste exemplo, em vez de utilizar opções de `hmmpfam`, utilize o nome da coluna apropriado na cláusula `WHERE` da consulta SQL.

A próxima tarefa nesta seqüência de tarefas é registrar pseudônimos para origens de dados HMMER.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados HMMER” na página 345

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Registrando Pseudônimos para Origens de Dados HMMER

O registro de pseudônimos para origens de dados HMMER faz parte da maior tarefa de adicionar um HMMER a um sistema federado. Depois de registrar o servidor, você deve registrar um pseudônimo correspondente. Ao referir-se a uma origem de dados HMMER em uma consulta, utilize pseudônimos.

Procedimento:

Para registrar um pseudônimo do HMMER, utilize a instrução `CREATE NICKNAME`. Você deve definir um pseudônimo separado para cada banco de dados do perfil que deseja consultar.

A sintaxe da instrução `CREATE NICKNAME` para o HMMER é:

```
►►—CREATE NICKNAME—pseudônimo—FOR SERVER—nome-do-servidor—OPTIONS—(—————►
►—DATASOURCE—'nome_da_origem_de_dados' —————►
└──,—TIMEOUT—'duração_do_tempo_limite' ───┘
```

```
┌,—HMMTYPE—'hmmmer_type' ─┐
└──────────────────────────┘
──────────────────────────┘
──────────────────────────┘
```

Exemplo:

```
CREATE NICKNAME myhmms
FOR SERVER hmmmer_server
OPTIONS(DATASOURCE 'myhmms', TIMEOUT '60')
```

Opções de Pseudônimos

Você deve colocar os valores de opções de pseudônimos entre aspas simples.

DATASOURCE

Obrigatório: O nome da origem de dados na qual você executa a pesquisa HMMER. A cadeia exata que é utilizada aqui deve estar presente no arquivo de configuração do daemon HMMER.

TIMEOUT

Opcional: O tempo máximo, em segundos, que o wrapper HMMER aguardará resultados do daemon. O valor padrão é 60.

HMMTYPE

Opcional: O alfabeto que é utilizado nos dois modelos e seqüências de genes. O valor pode ser NUCLEIC ou PROTEIN e não faz distinção entre maiúsculas e minúsculas. O valor padrão é PROTEIN.

Colunas Fixas

A instrução CREATE NICKNAME cria automaticamente colunas fixas. Você pode referir-se a estas colunas fixas em consultas SQL como parte da definição de pseudônimo. As colunas fixas não aparecem na instrução CREATE NICKNAME. Há dois tipos de colunas fixas, de entrada e de saída.

Colunas fixas de entrada

Utilize colunas fixas de entrada como predicados de transmissão de parâmetros em consultas SQL. Eles transmitem opções padrão hmmpfam para o HMMER. O HMMER é executado na origem de dados especificada que utiliza estas opções de hmmpfam. Você também pode referir-se a colunas fixas de entrada na lista de seleção de consultas, que são retornadas como parte da tabela de resultados. A Tabela 62 na página 347 lista as colunas fixas de entrada.

Tabela 62. Colunas fixas de entrada

| Nome | Tipo de dados | Descrição | Operadores Permitidos | Opção hmpfam | Valor Retornado |
|------------------|----------------|--|-----------------------|----------------------------------|--|
| HmmQSeq | varchar(32000) | Seqüência de genes utilizada para pesquisa | = | | Igual à entrada; esta coluna é obrig. |
| ModelEValue | double | E-value estimado | < | -E <i>n</i> | Consulte a saída. |
| ModelScore | double | Score bruto | > | -T <i>n</i> | Consulte a saída. |
| DBSize | integer | Calcular e-values como se o banco de dados tivesse seqüências de genes 'n' | = | -Z <i>n</i> | Igual à entrada; utiliza o padrão hmpfam se não for especific. |
| CutMode | char(2) | Modo de corte; pode ser ga, tc ou nc (distinção entre maiúsculas e minúsculas) | = | --cut_ga --cut_tc --cut_nc | Igual à entrada; NULL se não for especific. |
| DomainScore | double | Score de domínio | > | --domT <i>n</i> | Consulte a saída. |
| DomainEValue | double | E-value de domínio | < | --domE <i>n</i> | Consulte a saída. |
| ForwardAlgorithm | char | Utilizar algoritmo Forward em vez de Viterbi; o valor pode ser 'Y' ou 'N' | = | --forward | Igual à entrada; 'N' é o padrão. |

Colunas fixas de saída

Você pode utilizar as colunas fixas de saída que são retornadas na tabela de resultados da consulta como predicados. A Tabela 63 na página 348 lista colunas fixas de saída.

Tabela 63. Colunas fixas de saída

| Nome | Tipo de dados | Descrição |
|----------------------|----------------|--|
| Model | varchar(32) | Nome do modelo. |
| ModelDescription | varchar(64) | Descrição de texto do modelo. |
| ModelScore | double | Score bruto ("score de bit"). |
| ModelEValue | double | E-value estimado. |
| ModelHits | integer | Número de acertos de domínios no modelo. |
| DomainNumber | integer | Domínio específico (em um modelo). |
| SequenceFrom | integer | Ponto inicial de seqüência de genes. |
| SequenceFromGlobal | char | 'Y' se o alinhamento for iniciado no começo da seqüência de genes. |
| HmmFrom | integer | Ponto inicial do modelo de consenso. |
| HmmFromGlobal | char | 'Y' se o alinhamento for iniciado no começo do modelo de consenso. |
| HmmTo | integer | Ponto final no modelo de consenso. |
| HmmToGlobal | char | 'Y' se o alinhamento terminar no final do modelo de consenso. |
| DomainScore | double | Score bruto ("score de bit") para o domínio isolado. |
| DomainEValue | double | Valor esperado para o domínio isolado. |
| AlignmentConsensus | varchar(32000) | O consenso HMM (o aminoácido mostrado para o consenso é o aminoácido de maior probabilidade nessa posição, de acordo com o HMM, não necessariamente o aminoácido com score mais alto). |
| AlignmentExactMatch | varchar(32000) | Corresponde o resíduo de probabilidade mais alta no HMM. |
| AlignmentSubSequence | varchar(32000) | Mostra a própria seqüência de genes. |

Não há mais tarefas nesta seqüência de tarefas.

Related reference:

- "Origem de Dados HMMER – Exemplo Completo" na página 349
- "Instrução CREATE NICKNAME - Exemplo para o Wrapper HMMER" na página 349

Instrução CREATE NICKNAME - Exemplo para o Wrapper HMMER

A instrução CREATE NICKNAME a seguir define o pseudônimo myhmms:

```
CREATE NICKNAME myhmms
FOR SERVER hmmer_server
OPTIONS(DATASOURCE 'myhmms', TIMEOUT '60')
```

Depois de emitir a instrução CREATE NICKNAME, você pode utilizar o pseudônimo myhmms para consultar o sistema federado. Você também pode unir o pseudônimo myhmms a outros pseudônimos e tabelas no sistema federado.

Não há mais tarefas nesta seqüência de tarefas.

Origem de Dados HMMER – Exemplo Completo

As instruções SQL para origens de dados HMMER devem conter predicados de entrada especiais que são utilizados para transmitir opções HMMER padrão para o arquivo executável hmmpfam.

Restrições:

Para ser válida, cada consulta transmitida para o wrapper HMMER deve conter pelo menos o predicado de entrada HmMQSeq. Todos os outros predicados são opcionais.

Procedimento:

Para construir uma consulta HMMER, utilize os predicados de entrada na cláusula WHERE de sua instrução SQL.

O exemplo completo a seguir mostra todas as instruções necessárias para criar e executar uma consulta que utiliza HmMQSeq como uma seqüência de pesquisa:

```
CREATE WRAPPER hmmer_wrapper
LIBRARY 'libdb21shmmr.a'
OPTIONS (DB2_FENCED 'N');

CREATE SERVER hmmer_serv
TYPE pfam VERSION 2.2
WRAPPER hmmer_wrapper
OPTIONS(NODE 'HMMERserv.MyCompany.com');

CREATE NICKNAME myhmms
FOR SERVER hmmer_serv
OPTIONS(DATASOURCE 'myhmms', TIMEOUT '1');

-- Run the 7LES_DROME gene sequence on the myhmms nickname
```

```

SELECT Model, substr(ModelDescription,1,50) as ModelDescription,
       ModelScore, ModelEValue, ModelHits, DomainNumber,
       SequenceFrom, SequenceTo, SequenceFromGlobal, SequenceToGlobal,
       HmmFrom, HmmTo, HmmFromGlobal, HmmToGlobal,
       DomainScore, DomainEValue,
       length(HmmQSeq)      as "length(HmmQSeq)",
       length(AlignmentConsensus) as "length(AConsensus)",
       length(AlignmentMatch)   as "length(AMatch)",
       length(AlignmentSubSeq)  as "length(ASubSeq)",
       substr(HmmQSeq,1,64)      as HmmQSeq,
       substr(AlignmentConsensus,1,64) as AlignmentConsensus,
       substr(AlignmentMatch, 1,64) as AlignmentMatch,
       substr(AlignmentSubSeq, 1,64) as AlignmentSubSeq
FROM myhmms
WHERE HmmQSeq =
      'MTMFWQQNVDHQSDEQDKQAKGAAPT KRLNISFNVKI AVNVNTKMTTTHINQQAPGTSS...';

```

Related tasks:

- “Registando Pseudônimos para Origens de Dados HMMER” na página 345
- “Construir Novas Consultas HMMER com Amostras” na página 350

Construir Novas Consultas HMMER com Amostras

As consultas de amostra HMMER a seguir ilustram como construir consultas para origens de dados HMMER.

Procedimento:

Para executar consultas, utilize os exemplos a seguir como um guia.

Nestas consultas, o nome utilizado para cada pseudônimo descreve o tipo de pesquisa e origem de dados HMMER. Com nomes descritivos, não é necessário listar instruções de registro com cada consulta de amostra. Além disso, alguns exemplos ilustram o comportamento do wrapper HMMER quando unido a outras origens de dados.

Consulta 1.

```

SELECT Model, ModelScore, ModelEValue, DomainNumber, DomainScore, DomainEValue
FROM myhmms
WHERE HmmQSeq = 'MTMFWQQNVDHQSDEQDKQAKGAAPT KRLNISFNVKI AVNVNTKMTTTHINQ...'

```

Ao executar a instrução do SQL, o wrapper executará uma pesquisa hmmpfam de myhmms que utiliza a seqüência de genes indicada. O wrapper retornará todas as colunas disponíveis, incluindo as colunas de parâmetros de entrada e as colunas de resultados do HMMER.

Consulta 2.

```
SELECT Model, ModelScore, ModelEValue
FROM myhmms
WHERE HmmQSeq = 'MTMFWQNVNDHQSDEQDKQAKGAAPTkrlnISFNVKIAVNVNTKMTTTHINQ...'
AND ModelScore > 0
```

Ao executar a instrução do SQL, o wrapper executará uma pesquisa hmmpfam de myhmms que utiliza a sequência de genes indicada. Além disso, o wrapper transmite a opção -T 0 (consulte a Tabela 62 na página 347) para o comando hmmpfam. O wrapper retornará as três colunas listadas após SELECT.

Consulta 3.

```
SELECT Model, DomainNumber, DomainScore, DomainEValue
FROM myhmms
WHERE HmmQSeq = 'MTMFWQNVNDHQSDEQDKQAKGAAPTkrlnISFNVKIAVNVNTKMTTTHINQ...'
AND ModelEValue < 1
ORDER BY DomainScore DESC
```

Ao executar a instrução do SQL, o wrapper executará uma pesquisa hmmpfam de myhmms que utiliza a sequência de genes indicada. Além disso, o wrapper transmite a opção -E 1 (consulte a Tabela 62 na página 347) para o comando hmmpfam. O wrapper retornará as quatro colunas listadas após SELECT e classificará o resultado por DomainScore do maior para o menor.

Related tasks:

- “Origem de Dados Entrez — Consultas de Exemplo” na página 290

Related reference:

- “Origem de Dados de Documentum – Consultas de Exemplo” na página 188
- “Origem de Dados Excel – Consultas de Exemplo” na página 207
- “Wrapper Extended Search - Consultas de Exemplo” na página 323

Dicas de Otimização para o Wrapper HMMER

Executar o wrapper e o daemon no mesmo servidor pode eliminar possíveis gargalos na comunicação de rede.

Related tasks:

- “Definindo a Variável de Perfil DB2_DJ_COMM do DB2 para o Wrapper HMMER” na página 343

Related reference:

- “Dicas e Considerações de Otimização para o Wrapper do Arquivo Estruturado de Tabela” na página 160

- “Dicas de Otimização para o Wrapper do BLAST” na página 243

Mensagens para o Wrapper HMMER

Para que o wrapper HMMER funcione, é necessário especificar uma consulta que contenha um predicado na coluna HmmQSeq. Quando consultar um fragmento que não tenha um predicado na coluna HmmQSeq, você obterá um erro.

Esta seção lista e descreve mensagens que podem ser encontradas quando trabalhar com o wrapper HMMER.

Tabela 64. Mensagens do wrapper HMMER

| Código de Erro | Mensagem | Explicação |
|-----------------------|--|--|
| SQL0142N | A instrução SQL não é suportada. | A consulta SQL enviada para o DB2 não foi processada pelo wrapper. Adicione o predicado requerido e envie novamente. Verifique se o operador utilizado em um predicado é válido para essa coluna (consulte a Tabela 62 na página 347). |
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Hmmer wrapper". O texto e os tokens associados são "Impossível resolver o nome do host NODE". | O nome do NODE TCP/IP especificado em CREATE SERVER é inválido. |
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Hmmer wrapper". O texto e os tokens associados são "Não foi possível conectar ao daemon". | O programa hmmer_daemon não está em execução no nó de destino ou a DAEMON_PORT especificada no comando CREATE SERVER não corresponde ao valor DAEMON_PORT especificado no arquivo de configuração do daemon, HMMER_DAEMON.config. |
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Hmmer wrapper". O texto e os tokens associados são "Erro desconhecido do daemon hmmer". | O nome DATASOURCE especificado na instrução CREATE NICKNAME pode não corresponder a nenhum dos nomes de banco de dados do perfil listados no arquivo de configuração do daemon, HMMER_DAEMON.config. |

Tabela 64. Mensagens do wrapper HMMER (continuação)

| Código de Erro | Mensagem | Explicação |
|-----------------------|--|---|
| SQL1822N | Código de erro inesperado "Erro Não Especificado" recebido da origem de dados "Hmmer wrapper". O texto e os tokens associados são "FATAL: Não existe tal opção "--cut_TC". | O predicado CutMode deve ser especificado em letra minúscula. Exemplo: WHERE CutMode = 'tc' |

Related reference:

- "Mensagens para o Wrapper do Arquivo Estruturado de Tabela" na página 160
- "Mensagens para o Wrapper do Documentum" na página 194
- "Mensagens para o Wrapper do Excel" na página 211
- "Mensagens para o Wrapper do BLAST" na página 243
- "Mensagens para o Wrapper do XML" na página 268
- "Mensagens para o Wrapper Entrez" na página 300
- "Mensagens para o Wrapper Extended Search" na página 328

Capítulo 20. Alterando pseudônimos

Este capítulo explica como utilizar a instrução ALTER NICKNAME para alterar os pseudônimos registrados anteriormente.

Alterando Pseudônimos

Você pode usar a instrução ALTER NICKNAME para modificar a representação do banco de dados federado de uma origem de dados ou exibição.

Restrições:

A instrução ALTER NICKNAME não pode ser utilizada para alterar nomes de colunas para os wrappers BLAST, Documentum ou EXCEL. A instrução ALTER NICKNAME pode ser utilizada para alterar nomes de colunas para o arquivo estruturado de tabela e wrappers XML.

Procedimento:

Para alterar valores de colunas de pseudônimo, você deve utilizar a instrução ALTER NICKNAME:

- Alterar os tipos de dados locais destas colunas
- Incluir, alterar ou excluir opções destas colunas

Related tasks:

- “Alterando o Tipo de Dados” na página 355
- “Alterando a Opção de Pseudônimo” na página 356

Alterando o Tipo de Dados

Você pode utilizar a instrução ALTER NICKNAME para alterar o tipo de dados de uma coluna.

Procedimento:

Para alterar o tipo de dados de uma coluna, utilize a instrução ALTER NICKNAME.

Por exemplo, a instrução ALTER NICKNAME a seguir altera o tipo de dados local da coluna DRUG para CHAR(30). A coluna DRUG foi definida originalmente como um CHAR(20), utilizando uma instrução CREATE NICKNAME. O pseudônimo DRUGDATA1 refere-se a um arquivo estruturado de tabela local chamado drugdata1.txt.

```
ALTER NICKNAME DRUGDATA1
  ALTER COLUMN DRUG
  LOCAL TYPE CHAR(30)
```

Related tasks:

- “Alterando Pseudônimos” na página 355
- “Alterando a Opção de Pseudônimo” na página 356

Alterando a Opção de Pseudônimo

Você pode utilizar a instrução ALTER NICKNAME para alterar uma opção de pseudônimo.

Procedimento:

Para alterar uma opção de pseudônimo, utilize a instrução ALTER NICKNAME.

Por exemplo, a instrução ALTER NICKNAME a seguir altera o caminho completo para o arquivo estruturado de tabela, drugdata1.txt. O caminho foi definido inicialmente como '/user/pat/drugdata1.txt' usando uma instrução CREATE NICKNAME. O pseudônimo DRUGDATA1 refere-se a um arquivo estruturado de tabela local chamado drugdata1.txt.

```
ALTER NICKNAME DRUGDATA1
  OPTIONS (SET FILE_PATH '/usr/kelly/data/drugdata1.txt')
```

Related tasks:

- “Alterando Pseudônimos” na página 355
- “Alterando o Tipo de Dados” na página 355

Capítulo 21. Referência de Comandos DDL

Este capítulo fornece detalhes sobre as instruções de sintaxe, os argumentos e as opções para os comandos DDL do wrapper abrangidos por este manual. As instruções estão ordenadas alfabeticamente por instrução e depois por wrapper.

Sintaxe da Instrução CREATE FUNCTION - Wrapper Extended Search

```
►—CREATE FUNCTION ESWRAPPER.ES_SEARCH—(—INTEGER—,—VARCHAR(1024)—)—————►  
►—RETURNS INTEGER AS TEMPLATE—————►
```

INTEGER

Define o parâmetro de referência da consulta. Em uma consulta, este parâmetro deve especificar o nome de uma coluna INTEGER que está definida na tabela de pseudônimos para a qual esta função personalizada está sendo chamada. O valor deve ser uma coluna de ligação do pseudônimo, não uma constante (por exemplo, DOC_RANK).

O parâmetro de referência identifica o pseudônimo ao qual a função ES_SEARCH deve ser aplicada. O próprio parâmetro não é avaliado.

Se uma instrução SELECT contiver mais de uma tabela na cláusula FROM e a cláusula WHERE contiver uma instrução ES_SEARCH, o parâmetro de referência permitirá instruir o DB2 sobre a qual tabela uma instrução de pesquisa específica pertence. Por exemplo:

```
SELECT *  
FROM ES_Nickname_1 as N1, ES_Nickname_2 as N2  
WHERE ESWRAPPER.ES_SEARCH(N1.DOC_RANK, 'IBM')=1 AND  
      ESWRAPPER.ES_SEARCH(N2.DOC_RANK, 'LOTUS')=1
```

VARCHAR(1024)

Define a expressão de consulta. Em uma consulta, este parâmetro deve especificar uma cadeia que utiliza o generalized query language do Extended Search.

Related tasks:

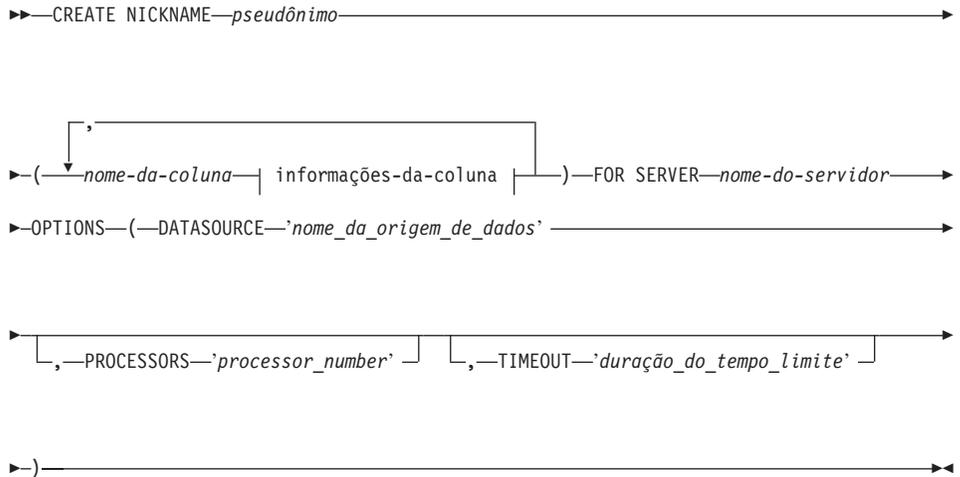
- “Registrando a Função Personalizada do Extended Search” na página 320

Related reference:

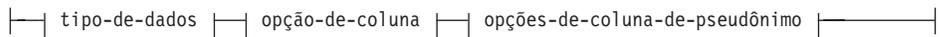
- “CREATE FUNCTION (Sourced or Template) statement” no *SQL Reference, Volume 2*

- “Wrapper Extended Search - Consultas de Exemplo” na página 323
- “Wrapper Extended Search - Linguagem de Consulta Generalizada” na página 326

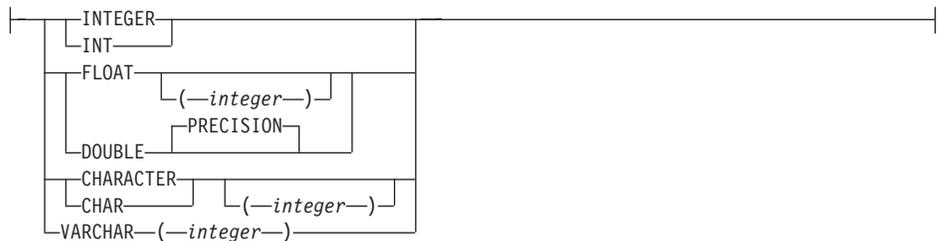
Sintaxe da Instrução CREATE NICKNAME - Wrapper BLAST



informações-da-coluna:



tipo-de-dados:



opção-de-coluna:



opções-de-coluna-de-pseudônimo:

```
|—OPTIONS—(—INDEX—'número_de_índice' —,—DELIMITER—'delimitador' —————→  
|—DEFAULT—'novo_valor_padrão' —————→
```

Opções de colunas de pseudônimos:

Os valores de opções de colunas de pseudônimo devem ser colocados entre aspas simples.

INDEX

O número ordinal da coluna na qual esta opção aparece no grupo de colunas de linha de definição. Esta opção é obrigatória.

DELIMITER

Os caracteres delimitadores que devem ser utilizados para determinar o ponto final das informações sobre linha de definição para a coluna na qual este tópico aparece. Se mais de um caractere aparecer no valor desta opção, a primeira ocorrência de qualquer um dos caracteres sinalizará o fim das informações sobre o campo. O padrão é o fim da linha. Esta opção é requerida, exceto para a última coluna especificada se você quiser que esta coluna contenha o restante da linha de definição.

DEFAULT

Especifica um novo valor padrão para as seguintes colunas fixas de entrada:

- E_value
- QueryStrands
- GapAlign
- NMisMatchPenalty
- NMatchReward
- Matrix
- FilterSequence
- NumberOfAlignments
- GapCost
- ExtendedGapCost
- WordSize
- ThresholdEx

Este novo valor substitui os valores padrão pré-definidos. O novo valor padrão deve ser do mesmo tipo que os valores indicados para uma determinada coluna. Esta opção é opcional.

Opções de pseudônimos:

Os valores das opções de pseudônimos devem ser colocados entre aspas simples.

DATASOURCE

O nome da origem de dados na qual a pesquisa do BLAST será executada. A cadeia exata utilizada aqui deve estar presente no arquivo de configuração do daemon do BLAST. Esta opção é obrigatória.

PROCESSORS

Especifica o número de processadores a serem utilizados ao avaliar uma consulta do BLAST. Corresponde à opção `blastall -a` do BLAST. Esta opção é opcional. O valor padrão é 1.

TIMEOUT

O tempo máximo, em minutos, que o wrapper do BLAST esperará para obter resultados do daemon. O padrão é 60. Esta opção é opcional.

Related tasks:

- “Configurando o Daemon do BLAST” na página 225
- “Registrando Pseudônimos para Origens de Dados do BLAST” na página 232

Related reference:

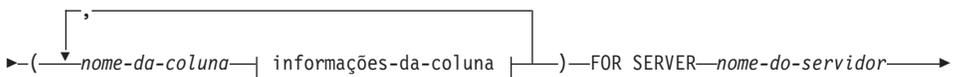
- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Instrução CREATE NICKNAME - Exemplos para Wrapper BLAST” na página 238

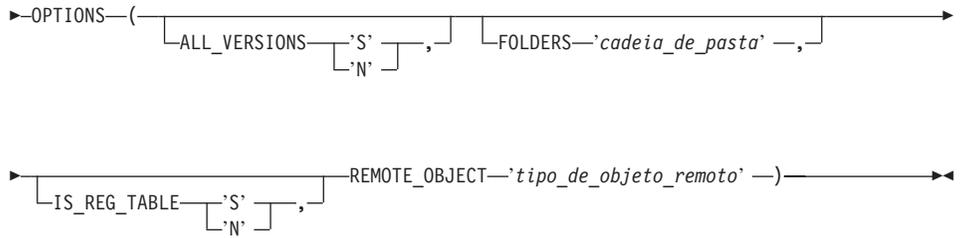
Sintaxe da Instrução CREATE NICKNAME - Wrapper Documentum

A sintaxe para a instrução CREATE NICKNAME do Documentum é:

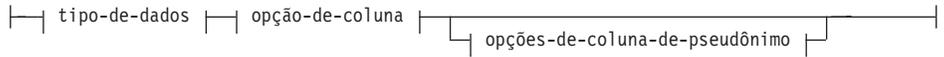
► CREATE NICKNAME *pseudônimo* ►

► (*nome-da-coluna* | informações-da-coluna |) FOR SERVER *nome-do-servidor* ►

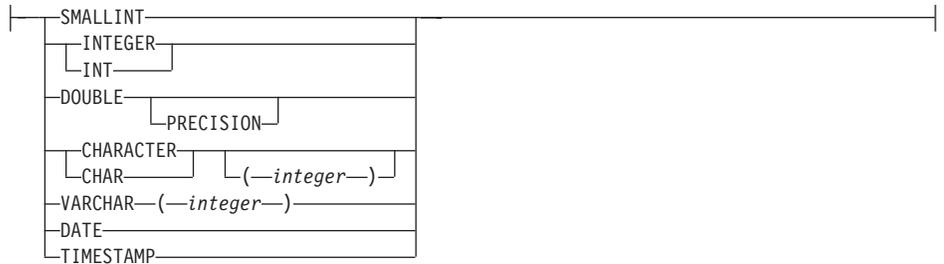




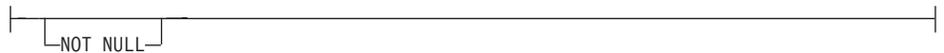
informações-da-coluna:



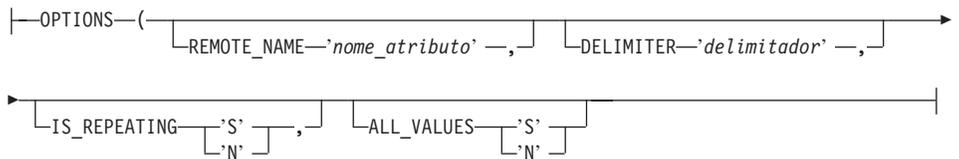
tipo-de-dados:



opção-de-coluna:



opções-de-coluna-de-pseudônimo:



As opções de colunas associadas à instrução CREATE NICKNAME para Documentum são:

NOT NULL

Todas as colunas de valor único exceto aquelas definidas como

TIMESTAMP e DATE devem ser definidas como NOT NULL. Os atributos de repetição não devem ser definidos como NOT NULL em pseudônimos.

As opções de colunas de pseudônimos associadas à instrução CREATE NICKNAME para Documentum são:

Os valores de opções de colunas de pseudônimo devem ser colocados entre aspas simples.

ALL_VALUES

Especifica que todos os valores de um atributo repetido serão retornados, separados pelo delimitador especificado. Se esta opção estiver faltando ou for 'N', apenas o último valor de um atributo repetido será retornado. Conforme foi notado sob DELIMITER, ALL_VALUES pode ser especificado apenas para colunas VARCHAR para as quais a opção IS_REPEATING for 'Y' (e é inválido quando IS_REG_TABLE = 'Y').

DELIMITER

Especifica a cadeia delimitadora a ser utilizada ao concatenar vários valores de um atributo de repetição. O delimitador pode ser um ou mais caracteres. O delimitador padrão é uma vírgula. Esta opção é válida somente para atributos de objetos com o tipo de dados VARCHAR, onde a opção IS_REPEATING está definida como 'Y'. Esta opção é opcional.

IS_REPEATING

Indica se a coluna possui vários valores. Os valores válidos são 'Y' e 'N'. O padrão é 'N'. Esta opção é opcional.

Apenas o último valor é retornado para

- atributos de repetição não-VARCHAR
- Colunas VARCHAR quando ALL_VALUES 'N' é especificado

Para superar essa limitação, você pode criar uma definição dupla para a coluna de atributos de repetição.

REMOTE_NAME

Especifica o nome do atributo ou da coluna correspondente do Documentum. Esta opção mapeia os nomes de atributos ou colunas remotos para os nomes de colunas locais do DB2. É padronizado para o nome da coluna do DB2. Esta opção é opcional.

As opções de colunas de pseudônimos associadas à instrução CREATE NICKNAME para Documentum são:

Os valores das opções de pseudônimos devem ser colocados entre aspas simples.

ALL_VERSIONS

Especifica se todas as versões do objeto serão pesquisadas. Os valores válidos são 'y', 'Y', 'n' e 'N'. O valor padrão de 'N' significa que apenas as versões atuais do objeto são incluídas no processamento de consulta. Esta opção não é válida quando IS_REG_TABLE = 'Y'. Esta opção é opcional.

FOLDERS

Especifica uma cadeia que contém um ou mais predicados FOLDER do Documentum combinados logicamente e corrigidos sintaticamente. A especificação dos predicados FOLDER restringe a definição de documentos representados por esse pseudônimo para aqueles existentes nas pastas designadas.

Ao especificar esta opção, coloque o valor inteiro da opção FOLDERS entre aspas simples e utilize aspas duplas no lugar das aspas simples dentro da cadeia.

Por exemplo, se você deseja inserir:

```
FOLDER('/Tools',DESCEND) OR FOLDER('/Cars')
```

Especifique a seguinte opção FOLDERS:

```
FOLDERS 'FOLDER("/Tools",DESCEND) OR FOLDER("/Cars")'
```

Esta opção não é válida quando IS_REG_TABLE = 'Y'. Esta opção é opcional.

IS_REG_TABLE

Especifica se o objeto especificado pela opção REMOTE_OBJECT é uma tabela registrada do Documentum. Os valores válidos são 'y', 'Y', 'n' e 'N'. O valor padrão é 'N'. Esta opção é opcional.

Você não pode alterar (ou voltar) um pseudônimo de um objeto do Documentum para uma tabela registrada alterando esta opção com a instrução ALTER NICKNAME. Ao invés disto, você deve utilizar as instruções DROP e re-CREATE para eliminar e criar novamente o pseudônimo.

REMOTE_OBJECT

Especifica o nome do tipo de objeto do Documentum associado ao pseudônimo. O nome pode ser qualquer tipo de objeto ou tabela registrada do Documentum. No caso de uma tabela registrada, ela deve ser prefixada pelo nome do proprietário da tabela. Se a tabela registrada pertencer ao proprietário do Docbase, dm_dbo poderá ser utilizado para o nome do proprietário. Esta opção é obrigatória.

FOR SERVER

Identifica o servidor que você registrou na instrução CREATE SERVER associada. Esse servidor é utilizado para acessar a planilha do Excel. Especifique o nome do servidor.

A lista a seguir descreve as opções CREATE NICKNAME para Excel:

FILE_PATH

Especifica o caminho do diretório e o nome do arquivo completamente qualificados da planilha do Excel que você deseja acessar.

Os tipos de dados devem ser consistentes dentro de cada coluna e os tipos de dados da coluna devem ser descritos corretamente durante o processo de registro do pseudônimo.

Os wrappers do Excel podem acessar apenas a planilha principal dentro de um livro de trabalho do Excel.

As células em branco na planilha são interpretadas como NULL.

Até 10 linhas em branco consecutivas podem existir na planilha e serem incluídas no conjunto de dados. Mais de 10 linhas consecutivas são interpretadas como o final do conjunto de dados.

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

A página de código do banco de dados deve corresponder ao conjunto de caracteres do arquivo; do contrário, você poderá obter resultados imprevistos.

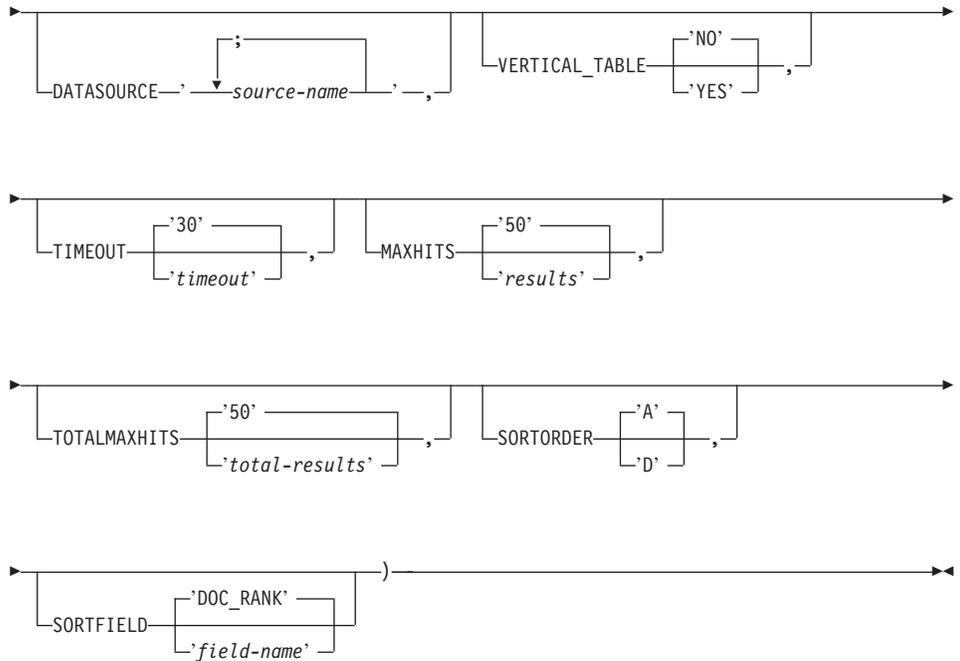
RANGE

Especifica um intervalo de células a serem utilizadas na origem de dados. Esta opção não é obrigatória.

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

- Os indicadores superior esquerdo e inferior direito não estão orientados corretamente. Uma orientação incorreta é aquela em que o indicador de célula superior esquerdo está abaixo ou à direita do indicador de célula inferior direito.
- O número de colunas designadas pelo valor de intervalo não corresponde ao número de colunas especificadas na instrução CREATE NICKNAME.
- Um caractere inválido ou outro erro de sintaxe foi localizado.

Aqui está um exemplo da opção de pseudônimo RANGE:



tipo-de-dados:



NICKNAME

Especifica um nome exclusivo para esta tabela de pseudônimos do Extended Search. Este nome deve ser distinto de todos os outros pseudônimos no esquema para o qual ele está sendo definido. Este parâmetro é obrigatório.

nome-da-coluna

Especifica um ou mais nomes de colunas definidos pelo usuário. O nome da coluna deve corresponder ao nome de um campo nativo ou mapeado que está definido no banco de dados de configuração do Extended Search. Este parâmetro é opcional.

tipo-de-dados

Especifica o tipo de dados de SQL da coluna denominada. Este tipo de dados deve corresponder ao tipo de dados definido para este campo no banco de dados de configuração do Extended Search. Por exemplo, para pesquisar um campo em uma origem de dados Extended Search que tenha um tipo de dados de Cadeia, defina uma coluna VARCHAR para este campo na tabela de pseudônimos. Se você especificar um *nome-da-coluna*, este parâmetro será obrigatório.

FOR SERVER

Especifica o nome de uma definição de servidor registrado anteriormente que foi criada para o servidor Extended Search que você deseja pesquisar. Este parâmetro é obrigatório.

APPLICATIONID

Especifica o nome do aplicativo Extended Search que você deseja pesquisar. Este nome deve existir no banco de dados de configuração do Extended Search. Este parâmetro é obrigatório.

CATEGORY

Especifica uma ou mais categorias do Extended Search que você deseja pesquisar. Se você omitir esta opção, deverá especificar pelo menos um nome de origem de dados. Para especificar várias categorias, delimite os nomes de categorias com um ponto e vírgula. Por exemplo:

```
CATEGORY 'LotusNotes;MSAccess;LDAP'
```

DATASOURCE

Especifica uma ou mais origens de dados Extended Search que deseja pesquisar. Se você omitir esta opção, deverá especificar pelo menos um nome de categoria. Para especificar várias origens de dados, delimite os nomes de origens de dados com um ponto e vírgula. Por exemplo:

```
DATASOURCE 'AltaVista;Google!;CNN'
```

VERTICAL_TABLE

Especifica o formato de apresentação para resultados da pesquisa. Se você especificar YES, o Extended Search retornará todos os campos que estão configurados como retornáveis, não apenas as colunas definidas pelo usuário. O wrapper armazena os resultados na tabela de pseudônimos como uma lista vertical de nomes de colunas. O valor padrão é NO.

TIMEOUT

Um INTEGER que especifica o número de segundos a aguardar por uma resposta de um servidor antes de exceder o tempo limite do pedido. Esta opção é opcional. O valor padrão é 30.

MAXHITS

Um INTEGER que especifica o número máximo de resultados que podem ser retornados de cada origem que está sendo pesquisada. Esta opção é opcional. O valor padrão é 50.

TOTALMAXHITS

Um INTEGER que especifica o número máximo de resultados que podem ser retornados de todas as origens que estão sendo pesquisadas. O wrapper combina estes resultados em um único conjunto de resultados. Esta opção é opcional. O valor padrão é 50.

SORTORDER

Especifica uma ordem de classificação para o retorno de resultados da pesquisa, em ordem crescente (A) ou decrescente (D). O valor padrão é A.

SORTFIELD

Especifica o nome de um campo no qual os resultados da pesquisa devem ser armazenados. O valor padrão, DOC_RANK, é um campo que o Extended Search utiliza para avaliar a importância de um documento de resultado. Se você especificar um nome de campo diferente, assegure que esse nome exista nas origens pesquisadas.

Related concepts:

- “Pseudônimos do Extended Search” na página 310
- “Tabelas Verticais do Extended Search” na página 312

Related tasks:

- “Registrando Pseudônimos para Origens de Dados Extended Search” na página 317

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Wrapper Extended Search - Consultas de Exemplo” na página 323

Opções da Instrução CREATE NICKNAME - Wrapper Entrez

A lista a seguir descreve as opções CREATE NICKNAME para Entrez:

REMOTE_OBJECT

Especifica o nome do tipo de objeto Entrez associado ao pseudônimo. Este nome determina o esquema e o banco de dados NCBI para o pseudônimo e seu relacionamento com outros pseudônimos. Este nome não faz distinção entre maiúsculas e minúsculas.

PARENT

Especificado apenas para um pseudônimo cujo pai foi renomeado através da opção REMOTE_OBJECT. A opção PARENT associa um filho a um pai quando várias famílias de pseudônimos estão definidas em um esquema do DB2. Este nome faz distinção entre maiúsculas e minúsculas.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados Entrez” na página 284

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*

Sintaxe da Instrução CREATE NICKNAME - Wrapper do Arquivo Estruturado de Tabela

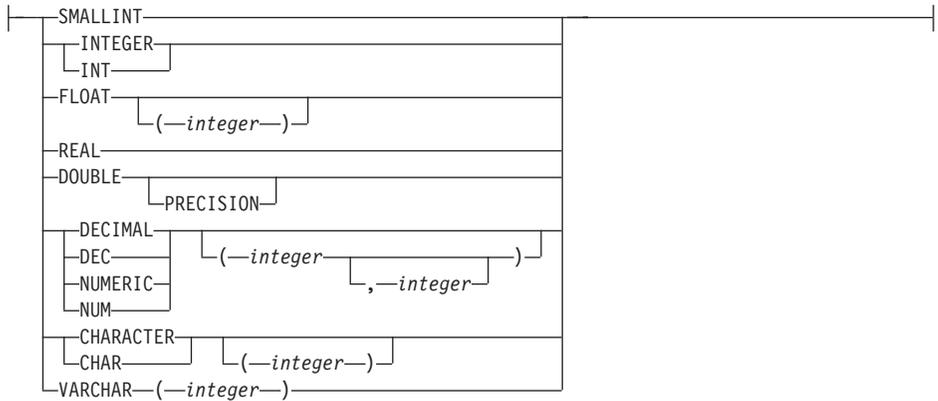
A sintaxe para a instrução CREATE NICKNAME é:



informações da coluna:

| tipo-de-dados | opção-de-coluna | opções da coluna de pseudônimo |

tipo-de-dados:



opção-de-coluna:



opções da coluna de pseudônimo:



Notas:

- 1 Não é permitido para arquivos não ordenados. Opcional para arquivos ordenados.

pseudônimo

Um pseudônimo exclusivo para o arquivo estruturado de tabela a ser acessado. Deve ser diferente de todos os outros pseudônimos, tabelas e exibições no esquema no qual está sendo registrado.

nome da coluna

Um nome exclusivo dado a cada campo no arquivo estruturado de tabela. Siga cada nome de coluna com seu tipo de dados. Somente colunas do tipo CHAR, VARCHAR, SMALLINT, INTEGER, FLOAT, DOUBLE, REAL e DECIMAL são suportadas.

SMALLINT

Para um número inteiro pequeno.

INTEGER ou INT

Para um número inteiro grande.

FLOAT(*inteiro*)

Para um número de precisão simples ou duplo do número de ponto flutuante, dependendo do valor do *inteiro*. O valor do *inteiro* deve

estar no intervalo de 1 a 53. Os valores de 1 a 24 indicam precisão simples e os valores de 25 a 53 indicam precisão dupla.

REAL Para vírgula flutuante de precisão simples.

DOUBLE ou **DOUBLE PRECISION**

Para vírgula flutuante de precisão dupla.

FLOAT

Para vírgula flutuante de precisão dupla.

DECIMAL(*precisão-do-inteiro, inteiro-da-escala*) ou **DEC**(*inteiro-da-precisão, inteiro-da-escala*)

Para um número decimal.

O primeiro inteiro é a precisão do número, ou seja, o número total de dígitos. Este valor pode ser de 1 a 31.

O segundo inteiro é a escala do número, ou seja, o número de dígitos à direita da vírgula decimal. Este valor pode ir de 0 à precisão do número.

Se a precisão e a escala não forem especificadas, os valores padrão de 5,0 serão usados.

As palavras **NUMERIC** e **NUM** podem ser usadas como sinônimo para **DECIMAL** e **DEC**.

CHARACTER(*inteiro*) ou **CHAR**(*inteiro*) ou **CHARACTER** ou **CHAR**

Para uma cadeia de caracteres de comprimento fixo de comprimento *inteiro*, que pode variar de 1 a 254. Se a especificação de comprimento for omitida, o comprimento de 1 caractere será assumido.

VARCHAR(*inteiro*)

Para uma cadeia de caracteres de comprimento variável com *inteiro* de comprimento máximo, que pode variar de 1 a 32672.

NOT NULL

Evita que a coluna contenha valores nulos.

O wrapper não impõe a limitação **NOT NULL**, mas o DB2 sim. Se você criar um pseudônimo e anexar uma limitação **NOT NULL** em uma coluna e, em seguida, selecionar uma linha contendo um valor nulo para a coluna, o DB2 emitirá um erro SQL0407N indicando que você não poderá atribuir um valor **NULL** a uma coluna **NOT NULL**.

A exceção a esta regra é para os pseudônimos ordenados. A coluna-chave dos pseudônimos ordenados não pode ser **NULL**. Se a coluna-chave **NULL** for encontrada para um pseudônimo ordenado, o erro SQL1822N será emitido, indicando que a coluna-chave está ausente.

FOR SERVER

Identifica o servidor que você registrou utilizando a instrução `CREATE SERVER`. Esse servidor será utilizado para acessar o arquivo estruturado de tabela.

FILE_PATH

O caminho completamente qualificado para o arquivo estruturado de tabela a ser acessado, colocado entre 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. A opção de coluna de pseudônimo `FILE_PATH` ou `DOCUMENT` deve ser especificada. Se a opção de pseudônimo `FILE_PATH` for especificada, nenhuma opção de coluna de pseudônimo `DOCUMENT` poderá ser especificada.

SORTED

Especifica se o arquivo de origem de dados é ordenado ou não. Esta opção também aceita 'Y', 'y', 'n' ou 'N'. Seu valor padrão é 'N'.

As origens de dados ordenadas devem ser classificadas na ordem crescente, de acordo com a seqüência de intercalação das características locais atuais, como indicado pelas definições na categoria do `LC_COLLATE National Language Support`.

Se você especificar que a origem de dados seja ordenada, recomendamos definir `VALIDATE_DATA_FILE` para 'Y'.

COLUMN_DELIMITER

O delimitador utilizado para separar colunas do arquivo estruturado de tabela, colocado entre aspas simples. Apenas delimitadores de caractere único são permitidos. Se nenhum delimitador de colunas estiver definido, será assumida a vírgula. Uma aspa simples não pode ser utilizada como um delimitador. O delimitador de colunas deve ser compatível em todo o arquivo. Um valor nulo é representado por dois delimitadores, um ao lado do outro ou um delimitador seguido por um terminador de linha, se o campo `NULL` for o último na linha. O delimitador de colunas não pode existir como sendo parte dos dados válidos para uma coluna. Por exemplo, um delimitador de colunas de vírgula não poderá ser usado se uma das colunas contiver dados com vírgulas embutidas.

KEY_COLUMN

O nome da coluna no arquivo que forma a chave na qual o arquivo é armazenado, colocado entre aspas simples. Use esta opção somente para arquivos ordenados. Uma coluna que é designada com a opção de coluna do pseudônimo `DOCUMENT` não deve ser especificada como uma coluna-chave.

Somente chaves da coluna simples são suportadas. Chaves de colunas múltiplas não são permitidas. O valor deve ser o nome de uma coluna

definida na instrução CREATE NICKNAME. A coluna deve ser ordenada em ordem crescente. Se o valor não for especificado para um pseudônimo ordenado, o padrão será a primeira coluna no arquivo de pseudônimos. Recomendamos que a coluna chave seja designada de maneira que não possa ser anulada, incluindo-se a opção NOT NULL em sua definição na instrução de pseudônimo.

Esta opção faz distinção entre maiúsculas e minúsculas. Entretanto, o DB2 dobra nomes de coluna para letras maiúsculas a menos que a coluna seja definida com aspas duplas.

VALIDATE_DATA_FILE

Para arquivos ordenados, esta opção especifica se o wrapper verificará se a coluna chave foi ordenada em ordem crescente e verifica chaves NULL. Os únicos valores válidos para esta opção são 'Y' ou 'N', colocados entre aspas simples. A verificação é feita uma vez na hora de registro. Se esta opção não foi especificada, não ocorrerá validação. Esta opção não será permitida se a opção de coluna do pseudônimo DOCUMENT for utilizada para o caminho do arquivo.

DOCUMENT

Especifica o tipo de arquivo estruturado de tabela. No momento, este wrapper suporta apenas FILE para esta opção. Apenas uma coluna pode ser especificada com a opção DOCUMENT por pseudônimo. A coluna associada à opção DOCUMENT deve ter o tipo de dados VARCHAR ou CHAR.

Ao utilizar a opção de coluna de pseudônimo DOCUMENT em vez da opção de pseudônimo FILE_PATH implica que o arquivo correspondente a este pseudônimo será fornecido durante a execução da consulta. Se a opção DOCUMENT tiver o valor "FILE", isto significa que o que será fornecido durante a execução da consulta será o caminho completo do arquivo cujo esquema corresponde à definição de pseudônimo para este pseudônimo. O exemplo CREATE NICKNAME a seguir ilustra a utilização da opção de coluna de pseudônimo DOCUMENT.

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name VARCHAR(16),
  address VARCHAR(30),
  id VARCHAR(16)
)
FOR SERVER file_server
```

A seguinte consulta, que especifica a localização do arquivo estruturado de tabela na cláusula WHERE, agora pode ser executada no pseudônimo customers:

```
SELECT name, address, id FROM customers
WHERE doc='/home/db2user/Customers.txt'
```

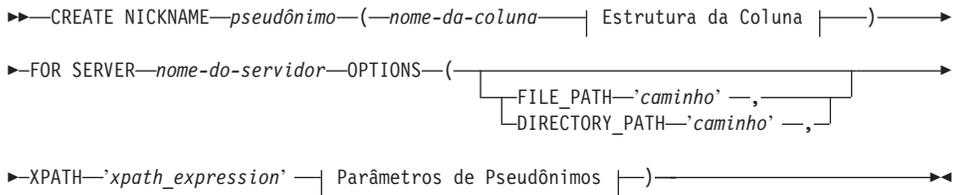
Related tasks:

- “Registrando o Servidor para Arquivos com Estrutura de Tabela” na página 156

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper do Arquivo Estruturado Tabela” na página 158

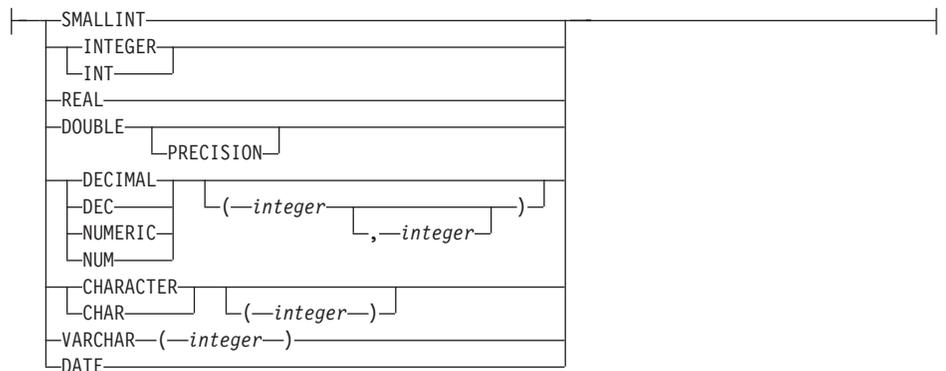
Sintaxe da Instrução CREATE NICKNAME - Wrapper XML



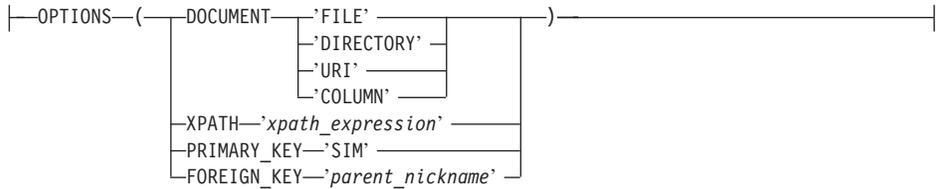
Estrutura da Coluna:



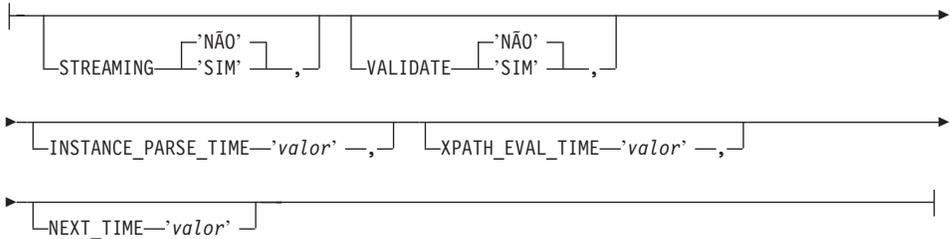
Opções de Tipos de Dados:



Opções de Colunas de Pseudônimos:



Parâmetros de Pseudônimos:



Parâmetros e Opções de Pseudônimos:

FILE_PATH

Especifica o caminho do arquivo do documento XML. Se você especificar esta opção de pseudônimo, não especifique uma coluna DOCUMENT. Esta opção FILE_PATH é aceita apenas para o pseudônimo raiz (o pseudônimo que identifica os elementos no nível superior do documento XML).

DIRECTORY_PATH

Especifica o nome do caminho de um diretório que contém um ou mais arquivos XML. Utilize esta opção para criar um único pseudônimo 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 demais arquivos neste diretório. Se você especificar esta opção de pseudônimo, não especifique uma coluna DOCUMENT. Esta opção DIRECTORY_PATH é aceita apenas para o pseudônimo raiz (o pseudônimo que identifica os elementos no nível superior do documento XML).

XPATH

Especifica uma expressão XPath que identifica os elementos XML que representam tuplas individuais. A opção de pseudônimo XPATH para um pseudônimo filho é avaliada no contexto do caminho especificado pela opção de pseudônimo XPATH de seu pai. Esta expressão XPath é utilizada como um contexto para avaliar os valores de colunas identificados pelas opções de colunas de pseudônimos XPATH.

Não especifique um prefixo de espaço de nomes em uma expressão XPath. O wrapper XML não suporta espaços de nomes.

Opções de colunas de pseudônimo:

DOCUMENT

Especifica que esta coluna é uma coluna DOCUMENT. O valor da coluna DOCUMENT indica o tipo de dados de origem XML fornecido para o pseudônimo quando a consulta é executada. Esta opção é aceita apenas para colunas do pseudônimo raiz (o pseudônimo que identifica os elementos no nível superior do documento XML). Apenas uma coluna pode ser especificada com a opção DOCUMENT por pseudônimo. A coluna que está associada à opção DOCUMENT deve ter um tipo de dados VARCHAR.

Se você utilizar uma opção de coluna DOCUMENT, em vez da opção FILE_PATH ou DIRECTORY_PATH, o documento que corresponde a este pseudônimo é fornecido quando a consulta é executada.

Os valores válidos para a opção DOCUMENT são:

FILE Especifica que o valor da coluna de pseudônimo está ligado ao nome do caminho de um arquivo que contém um documento XML. Os dados deste arquivo são fornecidos quando a consulta é executada.

DIRECTORY

Especifica que o valor da coluna de pseudônimo está ligado ao nome do caminho de um diretório que contém vários arquivos de dados XML. Os dados XML de vários arquivos são fornecidos quando a consulta é executada. Os dados estão localizados nos arquivos XML que residem no caminho do diretório especificado. O wrapper XML utiliza apenas os arquivos com uma extensão .xml que estão localizados no diretório especificado. O wrapper XML ignora todos os demais arquivos neste diretório.

URI Especifica que o valor da coluna de pseudônimo está ligado ao nome do caminho de um arquivo XML remoto ao qual uma URI se refere. O endereço do URI indica a localização remota deste arquivo XML na Web.

COLUMN

Especifica que o documento XML está armazenado em uma coluna relacional.

XPATH

Especifica a expressão XPath no documento XML que contém os dados correspondentes a esta coluna. O wrapper XML avalia a

expressão XPath depois que a instrução CREATE NICKNAME aplicar esta expressão XPath a partir desta opção de pseudônimo de XPATH.

Se você executar uma consulta em um nome de coluna que tenha uma referência de marcação XPATH configurada incorretamente, como um caso incorreto, a consulta retornará valores nulos nesta coluna para todas as linhas retornadas.

Não especifique um prefixo de espaço de nomes em uma expressão XPath. O wrapper XML não suporta espaços de nomes.

PRIMARY_KEY

Indica que este pseudônimo é um pseudônimo pai. O tipo de dados da coluna deve ser VARCHAR(16). Um pseudônimo pode ter, no máximo, uma opção de coluna PRIMARY_KEY. 'YES' é o único valor válido. A coluna que foi projetada com esta opção 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 para esta coluna. A coluna pode ser utilizada apenas para unir pseudônimos pai e pseudônimos filhos.

FOREIGN_KEY

Indica que este pseudônimo é um pseudônimo filho e especifica o nome do pseudônimo pai correspondente. Um pseudônimo pode ter, no máximo, uma opção de coluna FOREIGN_KEY. O valor para esta opção considera maiúsculas e minúsculas. A coluna que foi projetada com esta opção 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 para esta coluna. A coluna pode ser utilizada apenas para unir pseudônimos pai e pseudônimos filhos.

Uma instrução CREATE NICKNAME com uma opção FOREIGN_KEY falhará se o pseudônimo pai tiver um nome de esquema diferente.

A menos que o pseudônimo seja referido em uma cláusula FOREIGN_KEY que foi explicitamente definida para ter letras minúsculas ou maiúsculas e minúsculas, colocando-a entre aspas sob a instrução CREATE NICKNAME correspondente, quando referir-se a este pseudônimo na cláusula FOREIGN_KEY, será necessário especificar o pseudônimo em maiúscula.

Parâmetros de Pseudônimos:

STREAMING

Especifica se o documento de origem XML está separado em fragmentos lógicos que correspondem ao nó correspondente à expressão XPath do pseudônimo. O wrapper XML analisa e processa o fragmento de dados de origem XML por fragmento, reduzindo a

utilização total da memória. Você pode especificar streaming para qualquer documento de origem XML (FILE, DIRECTORY, URI ou COLUMN). Esta opção é aceita apenas para colunas do pseudônimo raiz (o pseudônimo que identifica os elementos no nível superior do documento XML). O valor de streaming padrão é NO.

Não defina o parâmetro STREAMING como YES se definir o parâmetro VALIDATE como YES. Se você definir ambos os parâmetros como YES, receberá uma mensagem de erro.

VALIDATE

Especifica se o documento de origem XML será validado antes da extração dos dados XML. Se esta opção for definida como YES, a opção de pseudônimo verificará se a estrutura do documento de origem está de acordo com um esquema XML ou a um DTD (document type definition). Esta opção é aceita apenas para colunas do pseudônimo raiz (o pseudônimo que identifica os elementos no nível superior do documento XML). O valor padrão é NO.

O documento de origem XML não será validado se o wrapper XML não puder localizar o arquivo de esquema XML ou arquivo DTD (.xsd ou .dtd). O DB2 não emite uma mensagem de erro se a validação não ocorrer. Portanto, certifique-se de que o arquivo de esquema XML ou o arquivo DTD exista na localização especificada no documento de origem XML.

Não defina o parâmetro VALIDATE como YES se definir o parâmetro STREAMING como YES. Se você definir ambos os parâmetros como YES, receberá uma mensagem de erro.

INSTANCE_PARSE_TIME

Especifica o tempo (em milissegundos) para analisar os dados em uma linha do documento de origem XML. Você pode modificar as opções INSTANCE_PARSE_TIME, XPATH_EVAL_TIME e NEXT_TIME para otimizar consultas de estruturas de origem XML grandes ou complexas. Esta opção é aceita apenas para colunas do pseudônimo raiz (o pseudônimo que identifica os elementos no nível superior do documento XML). O número especificado pode ser um inteiro ou um valor decimal. O valor padrão é de 7 milissegundos.

XPATH_EVAL_TIME

Especifica o tempo (em milissegundos) para avaliar a expressão XPath do pseudônimo e para localizar o primeiro elemento. Você pode modificar as opções XPATH_EVAL_TIME, INSTANCE_PARSE_TIME e NEXT_TIME para otimizar consultas de estruturas de origem XML grandes ou complexas. Esta opção é aceita para pseudônimos raiz e pseudônimos não raiz. O número especificado pode ser um inteiro ou um valor decimal. O valor padrão é 1 milissegundo.

NEXT_TIME

Especifica o tempo (em milissegundos) requerido para localizar elementos de origem subsequentes a partir da expressão XPath. Você pode modificar as opções NEXT_TIME, XPATH_EVAL_TIME e INSTANCE_PARSE_TIME para otimizar consultas de estruturas de origem XML grandes ou complexas. Esta opção é aceita para pseudônimos raiz e pseudônimos não raiz. O valor padrão é 1 milissegundo.

Notas sobre utilização:

Se você utilizar a opção de tipo de dados DATE, as datas no documento XML de origem deverão ter o seguinte formato: CCAA-MM-DD. Por exemplo, se a data for 17 de novembro de 2002, a data deverá ser especificada como 17/11/2002 no documento de origem XML. Se uma data tiver qualquer outro formato, você receberá uma mensagem de erro.

Não defina o parâmetro STREAMING e o parâmetro VALIDATE como YES. O wrapper XML valida um documento de origem XML inteiro e não valida fragmentos do documento de origem. Se você definir ambos os parâmetros como YES, receberá uma mensagem de erro.

Related tasks:

- “Registrando Pseudônimos para Origens de Dados XML” na página 260

Related reference:

- “CREATE NICKNAME statement” no *SQL Reference, Volume 2*
- “Instrução CREATE NICKNAME - Exemplos para o Wrapper XML” na página 261

Argumentos da Instrução CREATE SERVER - Wrapper BLAST

Argumentos:

TYPE Determina o tipo de pesquisa do BLAST realizada utilizando o servidor determinado. Este argumento é obrigatório. Ele deve ser definido para um dos seguinte valores: blastn, blastp, blastx, tblastn, tblastx.

VERSION

Especifica a versão do servidor que você está utilizando. Este valor deve ser definido para a versão do blastall que você está executando. Este argumento é obrigatório.

WRAPPER

Especifica o nome do wrapper que você registrou utilizando a instrução CREATE WRAPPER. Este argumento é obrigatório.

Opções:

NODE

Especifica o nome do host do sistema no qual o processo do daemon do BLAST está sendo executado. Esta opção é obrigatória.

DAEMON_PORT

Especifica o número da porta na qual o daemon escutará pedidos de job do BLAST. O número de porta deve ser o mesmo número especificado na opção `daemon_port` do arquivo de configuração do daemon. O padrão é 4007. Esta opção é opcional.

Related tasks:

- “Registrando um Servidor para uma Origem de Dados do BLAST” na página 231

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Argumentos e Opções da Instrução CREATE SERVER - Wrapper Documentum

Os argumentos associados à instrução CREATE SERVER para Documentum são:

TYPE Especifica o tipo da origem de dados. Para Documentum, o tipo é DCTM. Este argumento é obrigatório.

VERSION

Especifica a versão da origem de dados. Para EDMS98, o valor é '3'. Para 4i, o valor é '4'. Este argumento é obrigatório.

WRAPPER

Especifica o nome do wrapper associado a esse servidor. Este argumento é obrigatório.

As opções associadas à instrução CREATE SERVER para Documentum são:

CONTENT_DIR

Especifica o nome do diretório raiz localmente acessível para armazenar arquivos de conteúdo recuperados pelas pseudo colunas `GET_FILE`, `GET_FILE_DEL`, `GET_RENDITION` e `GET_RENDITION_DEL`. Ele deve ser gravado por todos os usuários capazes de utilizar estas pseudo colunas. Seu valor padrão é `/tmp`. Esta opção é opcional.

NODE

Especifica o nome real do Documentum Docbase. Esta opção é obrigatória.

OS_TYPE

Especifica o sistema operacional do servidor Docbase. Os valores válidos são AIX, SOLARIS e WINDOWS. Esta opção é obrigatória.

RDBMS_TYPE

Especifica o RDBMS utilizado pelo Docbase. Valores válidos são DB2, INFORMIX, ORACLE, SQLSERVER ou SYBASE. Esta opção é obrigatória.

TRANSACTIONS

Especifica o modo de transação do servidor. Os valores válidos são:

- NONE — nenhuma transação ativada.
- QUERY — as transações são ativadas apenas para os métodos Dctm_Query.
- ALL — as transações são ativadas para o método Dctm_Query. ALL tem a mesma função que QUERY neste release.

O padrão é QUERY. Esta opção é opcional.

Related tasks:

- “Registrando o Servidor para Origens de Dados do Documentum” na página 172

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Argumentos da Instrução CREATE SERVER - Wrapper Entrez

Os argumentos para a instrução CREATE SERVER para o Entrez são:

TYPE Especifica o tipo da origem de dados. Os valores aceitáveis para o tipo de servidor são PubMed e Nucleotide. Eles não fazem distinção entre maiúsculas e minúsculas.

VERSION

Especifica a versão do esquema XML NCBI que está sendo utilizada. Este argumento é opcional. Se a versão do servidor não for especificada, o padrão será 1.0.

WRAPPER

Especifica o nome do wrapper registrado utilizando a instrução CREATE WRAPPER.

Related tasks:

- “Registrando o Servidor para uma Origem de Dados Entrez” na página 283

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Argumentos da Instrução CREATE SERVER - Wrapper Excel

Os argumentos associados à instrução CREATE SERVER para Excel são:

WRAPPER

Especifica o nome do wrapper registrado na instrução CREATE WRAPPER associada. Este argumento é obrigatório.

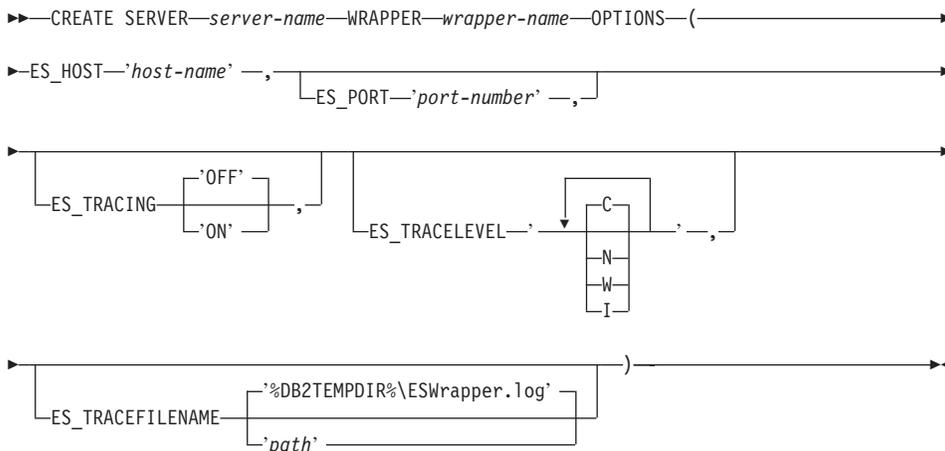
Related tasks:

- “Registrando o Servidor para uma Origem de Dados do Excel” na página 206

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Sintaxe da Instrução CREATE SERVER - Wrapper Extended Search



SERVER

Especifica um nome exclusivo para esta definição de servidor. Este parâmetro é obrigatório.

WRAPPER

Especifica o nome de um wrapper Extended Search registrado anteriormente que você deseja utilizar com esta definição de servidor. Este parâmetro é obrigatório.

ES_HOST

Especifica o nome completo do host ou endereço IP do servidor Extended Search que você deseja pesquisar. Esta opção é obrigatória.

ES_PORT

Especifica o número da porta na qual este servidor Extended Search atende pedidos. Se esta opção for omitida, o valor padrão será 6001.

ES_TRACING

Especifica se o rastreamento deve ser ativado para mensagens de erro, mensagens de aviso e mensagens informativas que são produzidas pelo servidor remoto Extended Search. O valor padrão, OFF, significa que não serão registradas mensagens de rastreamento.

ES_TRACELEVEL

Se o rastreamento estiver ativado, esta opção especificará os tipos de mensagens que serão gravadas no arquivo de log. O valor padrão, C, registra apenas mensagens críticas. Você pode ativar e desativar os seguintes níveis de rastreamento de forma independente:

- C – Mensagens de erro críticas
- N – Mensagens não críticas
- W – Mensagens de aviso
- I – Mensagens informativas

Por exemplo:

```
ES_TRACELEVEL 'W'  
ES_TRACELEVEL 'CN'  
ES_TRACELEVEL 'CNWI'
```

ES_TRACEFILENAME

Se o rastreamento estiver ativado, esta opção especificará o nome de um diretório e arquivo nos quais as mensagens serão gravadas. Se esta opção for omitida, o valor padrão será o arquivo ESWrapper.log no diretório temp (%DB2TEMPDIR%\ESWrapper.log do DB2 ou %DB2TEMPDIR%/ESWrapper.log).

Related tasks:

- “Registrando o Servidor para Origens de Dados Extended Search” na página 316

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Opções da Instrução CREATE USER MAPPING - Wrapper Documentum**Definições de opções:****REMOTE_AUTHID**

Seu identificador de autorização no servidor remoto.

REMOTE_PASSWORD

Sua senha no servidor remoto.

REMOTE_DOMAIN

Seu domínio de rede do Windows no servidor remoto. Válido apenas para plataformas Windows.

Related tasks:

- “Mapeando Usuários (Wrapper do Documentum)” na página 173

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Sintaxe da Instrução CREATE USER MAPPING - Wrapper Extended Search

```
►—CREATE USER MAPPING FOR—authorization-name—SERVIDOR—nome-do-servidor—►  
►—OPTIONS—(—REMOTE_AUTHID—'user-id' —,—REMOTE_PASSWORD—'password' —)—►◄
```

FOR

Especifica o ID do usuário de um usuário que você deseja autorizar para acessar origens de dados Extended Search. Este parâmetro é obrigatório.

SERVER

Especifica o nome de uma definição de servidor registrado anteriormente que foi criada para o servidor Extended Search que o usuário deseja pesquisar. Este parâmetro é obrigatório.

REMOTE_AUTHID

Especifica um ID do usuário que permite que este usuário do DB2 acesse origens de dados Extended Search. Este ID remoto deve estar no formato esperado pela origem de dados que está sendo pesquisada. Esta opção é obrigatória.

REMOTE_PASSWORD

Especifica a senha para este ID remoto. Esta opção é obrigatória.

Related tasks:

- “Registrando Mapeamentos de Usuários para Origens de Dados Extended Search” na página 318

Related reference:

- “CREATE USER MAPPING statement” no *SQL Reference, Volume 2*

Sintaxe da Instrução CREATE WRAPPER - Wrapper Extended Search

►►CREATE WRAPPER—*wrapper-name*—LIBRARY—'*library-name*' ◀◀

WRAPPER

Especifica um nome exclusivo para este wrapper Extended Search.

LIBRARY

Especifica um ou mais nomes de bibliotecas dependentes de plataforma:

- Windows: db2uies.dll
- AIX: libdb2uies.a

Related tasks:

- “Registrando o Wrapper Extended Search” na página 316

Related reference:

- “CREATE WRAPPER statement” no *SQL Reference, Volume 2*

Apêndice A. Exibições da Tabela de Catálogo Global Contendo Informações Federadas

A maioria das exibições de catálogo em um banco de dados federado são as mesmas que as exibições de catálogo de qualquer outro banco de dados DB2 para Linux, UNIX e Windows. Há várias exibições exclusivas que contêm informações pertinentes a um sistema federado, como a exibição SYSCAT.WRAPPERS.

Conforme observado nos manuais de Referência de SQL do DB2 para Linux, UNIX e Windows Versão 6 e Versão 7, as exibições DB2 Versão 8 SYSCAT agora são somente leitura. Se você emitir uma operação UPDATE ou INSERT em uma exibição no esquema SYSCAT, ela falhará. Utilizar as exibições SYSSTAT é a forma recomendada para atualizar o catálogo de sistema. Altere os aplicativos que fazem referência à exibição SYSCAT para fazer referência à exibição SYSSTAT atualizável.

A tabela a seguir lista as exibições SYSCAT que contêm informações federadas. Estas são exibições somente leitura.

Tabela 65. Exibições de catálogo geralmente utilizadas com um sistema federado

| Exibições do Catálogo | Descrição |
|------------------------------|---|
| SYSCAT.COLUMNS | Contém informações de colunas sobre os objetos de origem de dados (tabelas e exibições) para os quais você criou pseudônimos. |
| SYSCAT.COLOPTIONS | Contém informações sobre os valores de opções da coluna definidos para um pseudônimo. |
| SYSCAT.DATATYPES | Contém informações sobre os tipos de dados locais do DB2 internos e definidos pelo usuário. |
| SYSCAT.DBAUTH | Contém as autoridades do banco de dados mantidas por usuários individuais e grupos. |
| SYSCAT.FUNCMAPOPTIONS | Contém informações sobre os valores das opções definidos para o mapeamento de uma função. |
| SYSCAT.FUNCMAPPINGS | Contém os mapeamentos de funções entre o banco de dados federado e os objetos da origem de dados. |

Tabela 65. Exibições de catálogo geralmente utilizadas com um sistema federado (continuação)

| Exibições do Catálogo | Descrição |
|------------------------------|---|
| SYSCAT.ROUTINES | Contém funções locais definidas pelo usuário do DB2 ou gabaritos de funções. Os gabaritos de funções são utilizados para mapear para uma função da origem de dados. |
| SYSCAT.INDEXES | Contém especificações de índice para objetos de origem de dados. |
| SYSCAT.REVTYPEMAPPINGS | Contém mapeamentos de tipo de dados reverso. O mapeamento é a partir de tipos de dados locais do DB2 para tipos de dados de origem de dados. Estes mapeamentos são utilizados apenas com tabelas remotas (transparentes). |
| SYSCAT.SERVEROPTIONS | Contém informações sobre os valores de opção do servidor definidos com uma definição do servidor. |
| SYSCAT.SERVERS | Contém definições do servidor criadas para servidores da origem de dados. |
| SYSCAT.TABLES | Contém informações sobre cada tabela local do DB2, cada exibição federada e cada pseudônimo criado. |
| SYSCAT.TYPEMAPPINGS | Contém mapeamentos de tipo de dados de avanço. O mapeamento é para tipos de dados locais do DB2 a partir de tipos de dados de origem de dados. Estes mapeamentos são utilizados quando você consulta uma origem de dados utilizando uma instrução SQL do DB2. |
| SYSCAT.USEROPTIONS | Contém informações sobre autorização do usuário definidas quando você cria mapeamentos de usuários entre o banco de dados federado e os servidores de origem de dados. |
| SYSCAT.VIEWS | Contém informações sobre as exibições federadas criadas. |
| SYSCAT.WRAPOPTIONS | Contém informações sobre os valores de opções definidos para um wrapper. |
| SYSCAT.WRAPPERS | Contém o nome do wrapper e do arquivo de biblioteca para cada origem de dados para a qual um wrapper é criado. |

A tabela a seguir lista as exibições SYSSTAT que contêm informações federadas. Há exibições para leitura e gravação que contêm estatísticas que podem ser atualizadas.

Tabela 66. Exibições de catálogo globais federadas atualizáveis

| Exibições do Catálogo | Descrição |
|------------------------------|---|
| SYSSTAT.COLUMNS | Contém informações estatísticas sobre cada coluna dos objetos de origem de dados (tabelas e exibições) para os quais você criou pseudônimos. As estatísticas não são registradas para as colunas herdadas das tabelas digitadas. |
| SYSSTAT.FUNCTIONS | Contém informações estatísticas sobre cada função definida pelo usuário. Não inclui funções internas. As estatísticas não são registradas para as colunas herdadas das tabelas digitadas. |
| SYSSTAT.INDEXES | Contém informações estatísticas sobre cada especificação de índice para os objetos de origem de dados. |
| SYSSTAT.TABLES | Contém informações sobre cada tabela base. As informações sobre exibição, sinônimo e alias não estão incluídas nesta exibição. Para as tabelas digitadas, apenas a tabela raiz de uma hierarquia de tabelas é incluída na exibição. As estatísticas não são registradas para as colunas herdadas das tabelas digitadas. |

Apêndice B. Opções do Servidor para Sistemas Federados

As opções do servidor são utilizadas com a instrução CREATE SERVER para descrever um servidor de origem de dados. As opções do servidor especificam informações sobre a integridade dos dados, localização, segurança e desempenho. Algumas opções do servidor estão disponíveis para todas as origens de dados, e outras opções do servidor são específicas da origem de dados.

Wrappers não-relacionais têm opções adicionais bastante específicas do servidor que são documentadas nas informações de configuração da origem de dados.

As opções comuns do servidor federado para as origens de dados relacionais são:

- Opções de compatibilidade. COLLATING_SEQUENCE, IGNORE_UDT
- Opções de integridade dos dados. IUD_APP_SVPT_ENFORCE
- Opções de data e hora. DATEFORMAT, TIMEFORMAT, TIMESTAMPFORMAT
- Opções de localização. CONNECTSTRING, DBNAME, IFILE
- Opções de segurança. FOLD_ID, FOLD_PW, INFORMIX_LOCK_MODE
- Opções de desempenho. COMM_RATE, CPU_RATIO, DB2_MAXIMAL_PUSHDOWN, IO_RATIO, LOGIN_TIMEOUT, PACKET_SIZE, PLAN_HINTS, PUSHDOWN, TIMEOUT, VARCHAR_NO_TRAILING_BLANKS

A tabela a seguir lista as opções do servidor para definição do servidor aplicáveis para cada origem de dados relacional.

Tabela 67. Opções disponíveis do servidor

| Origem de Dados | COLLATING_SEQUENCE | COMM_RATE | CONNECTSTRING | CPU_RATIO | DATEFORMAT | DB2_MAXIMAL_PUSHDOWN | DBNAME | FOLD_ID | FOLD_PW | IFILE | IGNORE_UDT | INFORMIX_LOCK_MODE | IO_RATIO | IUD_APP_SVPT_ENFORCE | LOGIN_TIMEOUT | NODE | PACKET_SIZE | PASSWORD | PLAN_HINTS | PUSHDOWN | TIMEOUT | TIMEFORMAT | TIMESTAMPFORMAT | VARCHAR_NO_TRAILING_BLANKS |
|--------------------------------|--------------------|-----------|---------------|-----------|------------|----------------------|--------|---------|---------|-------|------------|--------------------|----------|----------------------|---------------|------|-------------|----------|------------|----------|---------|------------|-----------------|----------------------------|
| DB2 para iSeries | X | X | | X | | X | X | X | X | | | | X | X | | | | X | | X | | | | X |
| DB2 para z/OS e OS/390 | X | X | | X | | X | X | X | X | | | | X | X | | | | X | | X | | | | X |
| DB2 para VM e VSE | X | X | | X | | X | X | X | X | | | | X | X | | | | X | | X | | | | X |
| DB2 para Linux, UNIX e Windows | X | X | | X | | X | X | X | X | | | | X | X | | | | X | | X | | | | X |
| Informix | X | X | | X | | X | X | X | X | | X | X | X | X | | X | | X | | X | | | | |
| Microsoft SQL Server | X | X | | X | | X | X | X | X | | | | X | X | | X | | X | | X | | | | |
| ODBC | X | X | | X | X | X | X | X | X | | | | X | X | | X | | X | | X | | X | X | |
| OLE DB | X | | X | | | X | | | | | | | | | | | | | | | | | | |
| Oracle | X | X | | X | | X | X | X | X | | | | X | | | X | | X | X | X | | | | X |
| Sybase | X | X | | X | | X | X | X | X | X | X | | X | | X | X | X | X | X | X | X | | | |
| Teradata | X | X | | X | | X | | | | | | | X | X | | X | | | | X | | | | |

A tabela a seguir descreve cada opção do servidor e lista as definições válidas e padrão.

Tabela 68. Opções do servidor e suas definições

| Opção | Descrição e definições válidas | Definição padrão |
|--------------------|---|------------------|
| COLLATING_SEQUENCE | <p>Especifica se a origem de dados utiliza a mesma seqüência de intercalação padrão que o banco de dados federado, com base no conjunto de códigos NLS e nas informações sobre país/região.</p> <p>'Y' A origem de dados tem a mesma seqüência de intercalação que o banco de dados federado DB2.</p> <p>'N' A origem de dados tem uma seqüência de intercalação diferente da seqüência de intercalação do banco de dados federado do DB2.</p> <p>'T' A origem de dados tem uma seqüência de intercalação diferente da seqüência de intercalação do banco de dados federado do DB2 e a seqüência de intercalação da origem de dados não faz distinção entre maiúsculas e minúsculas (por exemplo, 'STEWART' e 'StewART' são considerados iguais).</p> | 'N' |
| COMM_RATE | <p>Especifica a taxa de comunicação entre o servidor federado e o servidor da origem e dados. Expressa em megabytes por segundo.</p> <p>Os valores válidos são maiores que 0 e menores que 1x1023. Os valores podem ser expressos em uma notação dupla válida, por exemplo, 123E10, 123 ou 1,21E4.</p> <p>Os valores podem ser expressos como números inteiros ou como números com ponto flutuante.</p> | '2' |
| CONNECTSTRING | Especifica as propriedades de inicialização necessárias para conectar a um provedor OLE DB. | Nenhuma |

Tabela 68. Opções do servidor e suas definições (continuação)

| Opção | Descrição e definições válidas | Definição padrão |
|---|--|------------------|
| CPU_RATIO | <p>Indica o quanto uma CPU de origem de dados é executada mais rapidamente ou mais lentamente do que a CPU do servidor federado.</p> <p>Os valores válidos são maiores que 0 e menores que 1x1023. Os valores podem ser expressos em uma notação dupla válida, por exemplo, 123E10, 123 ou 1,21E4.</p> <p>Os valores podem ser expressos como números inteiros ou como números com ponto flutuante.</p> <p>Uma definição igual a 1 indica que a velocidade da CPU federada do DB2 e a velocidade da CPU da origem de dados têm a mesma velocidade de CPU, uma proporção de 1:1. Uma definição igual a .5 indica que a velocidade da CPU federada do DB2 é 50% mais lenta do que a velocidade da CPU da origem de dados. Uma definição igual a 2 indica que a velocidade da CPU federada do DB2 é duas vezes mais rápida do que a velocidade da CPU da origem de dados.</p> | '1.0' |
| DATEFORMAT (Consulte a nota 5 no final desta tabela) | <p>O formato de data utilizado pela origem de dados. Insira o formato utilizado 'DD', 'MM' e 'YY' ou 'YYYY' para representar a forma numérica da data. Você também deve especificar o delimitador, como um espaço ou vírgula. Por exemplo, para representar o formato de data para '2003-01-01', utilize 'YYYY-MM-DD'. Este campo pode ser anulado.</p> | Nenhuma |

Tabela 68. Opções do servidor e suas definições (continuação)

| Opção | Descrição e definições válidas | Definição padrão |
|----------------------|---|------------------|
| DB2_MAXIMAL_PUSHDOWN | <p>Especifica os principais critérios que o otimizador de consulta utiliza ao escolher um plano de acesso. O otimizador de consulta pode escolher planos de acesso com base em custo ou no requisito do usuário de que seja processado o maior número de consultas possível pelas origens de dados remotas.</p> <p>'Y' O otimizador de consulta escolhe um plano de acesso que efetua pushdown de mais operações de consulta para a origem de dados do que outros planos. Quando vários planos de acesso fornecem a mesma quantidade de pushdown, o otimizador de consulta escolhe o plano com o menor custo.</p> <p>Se uma MQT (Materialized Query Table) do servidor federado puder processar parte da consulta ou toda ela, um plano de acesso que inclua a MQT poderá ser utilizado. As consultas que resultam em um produto Cartesiano serão processadas pelo banco de dados federado e não sofrerão pushdown.</p> <p>'N' O otimizador de consulta escolhe um plano de acesso com base em custo.</p> | 'N' |
| DBNAME | <p>Nome do banco de dados da origem de dados que você deseja que o servidor federado acesse. Para o DB2, este valor corresponde a um banco de dados específico de uma instância ou, com o DB2 para z/OS ou OS/390, o valor LOCATION do banco de dados. Não se aplica a origens de dados do Oracle, porque as instâncias Oracle contêm apenas um banco de dados.</p> | Nenhuma. |

Tabela 68. Opções do servidor e suas definições (continuação)

| Opção | Descrição e definições válidas | Definição padrão |
|--|---|------------------|
| FOLD_ID (Consulte as notas 1 e 4 no final desta tabela.) | <p>Aplica-se a IDs de usuários que o servidor federado envia ao servidor da origem de dados para autenticação. Os valores válidos são:</p> <p>'U' O servidor federado transforma o ID do usuário em letras maiúsculas antes de enviá-lo à origem de dados. Esta é uma opção lógica para a família DB2 e para as origens de dados do Oracle (Consulte a nota 2 no final desta tabela.)</p> <p>'N' O servidor federado não faz nada com o ID do usuário antes de enviá-lo à origem de dados. (Consulte a nota 2 no final desta tabela.)</p> <p>'L' O servidor federado transforma o ID do usuário em letras minúsculas antes de enviá-lo à origem de dados.</p> <p>Se nenhuma dessas definições for utilizada, o servidor federado tenta enviar o ID do usuário para a origem de dados em maiúsculas. Se o ID do usuário falhar, o servidor tentará enviá-lo em minúsculas.</p> | Nenhuma. |
| FOLD_PW (Consulte as notas 1, 3 e 4 no final desta tabela.) | <p>Aplica-se a senhas que o servidor federado envia às origens de dados para autenticação. Os valores válidos são:</p> <p>'U' O servidor federado transforma a senha em letras maiúsculas antes de enviá-la à origem de dados. Esta é uma opção lógica para a família DB2 e para as origens de dados do Oracle.</p> <p>'N' O servidor federado não faz nada com a senha antes de enviá-la à origem de dados.</p> <p>'L' O servidor federado transforma a senha em letras minúsculas antes de enviá-la à origem de dados.</p> <p>Se nenhuma dessas definições for utilizada, o servidor federado tentará enviar a senha para a origem de dados em maiúsculas. Se a senha falhar, o servidor tentará enviá-la em minúsculas.</p> | Nenhuma. |
| IFILE | <p>Especifica o caminho e o nome do arquivo de interface do Sybase Open Client. Nos servidores federados Windows NT, o padrão é %DB2PATH%\interfaces. Nos servidores federados UNIX, o caminho e o nome padrão são \$DB2INSTANCE/sqlib/interfaces.</p> | Nenhuma. |

Tabela 68. Opções do servidor e suas definições (continuação)

| Opção | Descrição e definições válidas | Definição padrão |
|--------------------|---|------------------|
| IGNORE_UDT | <p>Especifica se os UDTs (User-defined Types) das origens de dados acessadas utilizando o wrapper CTLIB ou DBLIB devem ser utilizadas pelo servidor federado. Os valores válidos são:</p> <p>'Y' Ignorar especificações definidas pelo usuário de UDTs.</p> <p>'N' Não ignorar especificações definidas pelo usuário de UDTs.</p> | 'N' |
| INFORMIX_LOCK_MODE | <p>Especifica o modo travar para ser definido para uma origem de dados do Informix. O wrapper do Informix emite o comando 'SET LOCK MODE' imediatamente depois de estabelecer a conexão a uma origem de dados do Informix. Os valores válidos são:</p> <p>'W' Define o modo travar do Informix para WAIT. Se o wrapper tentar acessar uma tabela ou linha travada, o Informix aguardará até a trava ser liberada.</p> <p>'N' Define o modo travar do Informix para NOWAIT. Se o wrapper tentar acessar uma tabela ou linha travada, o Informix retornará um erro.</p> <p>'n' Define o modo travar do Informix para WAIT <i>n</i> segundos. Se o wrapper tentar acessar uma tabela ou linha travada e a trava não for liberada dentro do número de segundos especificado, o Informix retornará um erro.</p> | 'W' |
| IO_RATIO | <p>Denota o quanto um sistema de E/S de origem de dados é executado mais rapidamente ou mais lentamente do que o sistema de E/S do servidor federado.</p> <p>Os valores válidos são maiores que 0 e menores que 1×10^{23}. Os valores podem ser expressos em uma notação dupla válida, por exemplo, 123E10, 123 ou 1,21E4.</p> <p>Uma definição igual a 1 indica que a velocidade de E/S federada do DB2 e a velocidade de E/S da origem de dados têm a mesma velocidade de E/S, uma proporção de 1:1. Uma definição igual a .5 indica que a velocidade de E/S federada do DB2 é 50% mais lenta do que a velocidade de E/S da origem de dados. Uma definição igual a 2 indica que a velocidade de E/S federada do DB2 é duas vezes mais rápida do que a velocidade de E/S da origem de dados.</p> | '1.0' |

Tabela 68. Opções do servidor e suas definições (continuação)

| Opção | Descrição e definições válidas | Definição padrão |
|----------------------|--|------------------|
| IUD_APP_SVPT_ENFORCE | <p>Especifica se o sistema federado do DB2 deve reforçar a detecção ou construção de instruções de salvamento de aplicativos. Quando definida através da instrução SET SERVER OPTION, esta opção do servidor não terá nenhum efeito com instruções SQL estáticas.</p> <p>'Y' O servidor federado irá retroceder transações de inserção, atualização ou exclusão se ocorrer um erro e a origem de dados não reforçar as instruções de salvamento de aplicativos. Um código de erro SQL (SQL1476) é retornado.</p> <p>'N' O servidor federado não retrocederá transações quando um erro for encontrado. Seu aplicativo deverá tratar da recuperação do erro.</p> | 'S' |
| LOGIN_TIMEOUT | Especifica o número de segundos que o servidor federado DB2 deve aguardar por uma resposta do Sybase Open Client para um pedido de login. Os valores padrão são os mesmos que para TIMEOUT. | '0' |
| NODE | Nome pelo qual uma origem de dados é definida como uma instância para seu RDBMS. | Nenhuma. |
| PACKET_SIZE | Especifica o tamanho do pacote do arquivo de interfaces do Sybase em bytes. Se a origem de dados não suportar o tamanho do pacote especificado, a conexão falhará. Aumentar o tamanho do pacote quando cada registro é muito grande (por exemplo, quando for inserir linhas em tabelas grandes) aumenta de forma significativa o desempenho. O tamanho em bytes é um valor numérico. | |
| PLAN_HINTS | <p>Especifica se as dicas de plano devem ser ativadas. As dicas de plano são fragmentos de instruções que fornecem informações extras para os otimizadores da origem de dados. Essas informações podem, para determinados tipos de consultas, melhorar o desempenho da consulta. As dicas de plano podem ajudar o otimizador da origem de dados a decidir se deve utilizar um índice, qual índice utilizar ou qual seqüência de junções de tabela utilizar.</p> <p>'Y' As dicas de plano devem ser ativadas na origem de dados se a origem de dados suportar as dicas de plano.</p> <p>'N' As dicas de plano não devem ser ativadas na origem de dados.</p> <p>Esta opção está disponível apenas para origens de dados do Oracle e Sybase.</p> | 'N' |

Tabela 68. Opções do servidor e suas definições (continuação)

| Opção | Descrição e definições válidas | Definição padrão |
|--|--|------------------|
| PUSHDOWN | <p>'Y' O DB2 irá considerar a hipótese de permitir que a origem de dados avalie as operações.</p> <p>'N' O DB2 enviará as instruções SQL da origem de dados que incluem apenas SELECT com nomes de colunas. Coluna de predicados (como WHERE=) e funções escalares (como MAX e MIN), ordenações (como ORDER BY ou GROUP BY) e junções não serão incluídas em nenhuma instrução SQL enviada à origem de dados.</p> | 'S' |
| TIMEFORMAT (Consulte a nota 5 no final desta tabela) | O formato de hora utilizado pela origem de dados. Insira o formato utilizando 'hh12', 'hh24', 'mm', 'ss', 'AM' ou 'A.M'. Por exemplo, para representar o formato de hora '16:00:00', utilize 'hh24:mm:ss'. Para representar o formato de hora '8:00:00 AM', utilize 'hh12:mm:ss AM'. Este campo pode ser anulado. | Nenhuma |
| TIMESTAMPFORMAT (Consulte a nota 5 no final desta tabela) | O formato de data e hora utilizado pela origem de dados. O formato segue o mesmo para data e hora, mais 'n' para décimos de segundos, 'nn' para centésimos de segundos, 'nnn' para milissegundos e assim por diante, até 'nnnnnn' para microssegundos. Por exemplo, para representar o formato de data e hora '2003-01-01-24:00:00.000000', utilize 'YYYY-MM-DD-hh24:mm:ss.nnnnnn'. Este campo pode ser anulado. | Nenhuma |
| TIMEOUT | Especifica o número de segundos que o servidor federado do DB2 irá aguardar por uma resposta do Sybase Open Client para qualquer instrução SQL. O valor de <i>segundos</i> é um número inteiro positivo no intervalo de inteiros do DB2 Universal Database. O valor do tempo limite especificado depende de qual wrapper você está utilizando. O comportamento padrão da opção TIMEOUT para os wrappers do Sybase é 0, que faz com que o DB2 aguarde indefinidamente por uma resposta. | '0' |

Tabela 68. Opções do servidor e suas definições (continuação)

| Opção | Descrição e definições válidas | Definição padrão |
|----------------------------|--|------------------|
| VARCHAR_NO_TRAILING_BLANKS | <p>Esta opção se aplica às origens de dados com tipos de dados de caracteres de variáveis que não têm o comprimento preenchido por espaços em branco à direita.</p> <p>Algumas origens de dados, como o Oracle, não têm semântica de comparação de caracteres de preenchimento em branco que retornam os mesmos resultados que a semântica de comparação do DB2 para Linux, UNIX e Windows. Defina esta opção quando quiser que ela se aplique a todas as colunas VARCHAR e VARCHAR2 dos objetos da origem de dados que serão acessadas a partir do servidor designado. Isto inclui exibições.</p> <p>A única definição válida para as origens de dados da família DB2 é 'Y'.</p> <p>'Y' Sim, espaços em branco à direita não existem nestas colunas VARCHAR.</p> <p>Esta origem de dados tem semântica de comparação de caracteres de preenchimento em branco diferente da do servidor federado. As operações de comparação de caracteres serão processadas no servidor federado e não sofrerão pushdown para a origem de dados.</p> <p>'N' Não, espaços em branco à direita estão presentes nestas colunas VARCHAR.</p> <p>Esta origem de dados tem semântica de comparação de caracteres de preenchimento em branco similar à semântica do servidor federado. As operações de comparação de caracteres podem sofrer pushdown para a origem de dados para processamento.</p> | 'N' |

Notas desta tabela:

1. Este campo é aplicado independente do valor especificado para autenticação.
2. Como o DB2 armazena IDs de usuários em letras maiúsculas, os valores 'N' e 'U' são logicamente equivalentes um ao outro.
3. A definição para FOLD_PW não tem nenhum efeito quando a definição para a senha for 'N'. Como nenhuma senha é enviada, maiúscula ou minúscula não pode ser um fator.
4. Evite definições nulas para qualquer uma dessas opções. Uma definição nula pode parecer atraente, porque o DB2 fará várias tentativas para resolver IDs de usuários e senha; no entanto, o desempenho pode ser

prejudicado (é possível que o DB2 envie um ID de usuário e uma senha quatro vezes antes de passar pela autenticação da origem de dados com êxito).

5. Esta opção é utilizada apenas quando o valor de `SERVER_TYPE` for `GENERIC`. Esta opção será ignorada para todos os outros valores de `SERVER_TYPE`

Related concepts:

- “Server characteristics affecting pushdown opportunities” no *Federated Systems Guide*
- “Server characteristics affecting global optimization” no *Federated Systems Guide*

Related reference:

- “CREATE SERVER statement” no *SQL Reference, Volume 2*

Apêndice C. Opções de Mapeamento de Usuário para Sistemas Federados

As opções de mapeamento de usuário fornecem informações sobre autorização e sobre a cadeia de contabilidade para mapeamentos de usuários entre o servidor federado e uma origem de dados. Estas opções podem ser utilizadas com qualquer origem de dados que suporte autorização de ID do usuário e senha.

Estas opções são utilizadas com a instrução CREATE USER MAPPING.

Tabela 69. Opções de mapeamento de usuário e suas definições

| Opção | Definições válidas | Definição padrão |
|-------------------|---|---|
| ACCOUNTING_STRING | Utilizada para especificar uma cadeia de contabilidade DRDA. As definições válidas incluem qualquer cadeia com comprimento de 255 ou menos. Esta opção é requerida apenas se as informações de contabilidade precisarem ser transmitidas. Consulte o Guia do Usuário do DB2 Connect para obter informações adicionais. | Nenhuma |
| REMOTE_AUTHID | Indica o ID de autorização utilizado na origem de dados. As definições válidas incluem qualquer cadeia com comprimento de 255 ou menos. | O ID utilizado para conectar ao banco de dados federado |
| REMOTE_DOMAIN | Indica o domínio do Windows NT utilizado para autenticar usuários que estejam conectando a uma origem de dados do Documentum. As definições válidas incluem qualquer nome de domínio válido do Windows NT. Se esta opção não for especificada, a origem de dados do Documentum autenticará utilizando o domínio de autenticação padrão para o banco de dados. | Nenhuma |
| REMOTE_PASSWORD | Indica a senha de autorização utilizada na origem de dados. As definições válidas incluem qualquer cadeia com comprimento 32 ou inferior. Se esta opção não for especificada, nenhuma senha será utilizada para conectar ao servidor da origem de dados. Se o servidor requerer uma senha para conectar, a conexão falhará. | Nenhuma |

Related concepts:

- “DB2 Connect and DRDA” no *DB2 Connect User’s Guide*

- “DRDA and data access” no *DB2 Connect User’s Guide*

Apêndice D. Opções de Coluna para Sistemas Federados

Você pode especificar informações sobre coluna nas instruções CREATE NICKNAME ou ALTER NICKNAME utilizando parâmetros chamados *opções de coluna*.

A principal finalidade das opções de coluna é fornecer informações sobre as colunas de pseudônimos para o SQL Compiler. Definir as opções de coluna de uma ou mais colunas para 'Y' permite que o SQL Compiler considere possibilidades adicionais de pushdown para predicados que executem operação de avaliação. Isto auxilia o Compiler na obtenção de otimização global. Você pode especificar qualquer um desses valores em caracteres maiúsculos ou minúsculos.

Atenção: Wrappers não-relacionais permitem opções de colunas adicionais.

Tabela 70. Opções de coluna e suas definições

| Opção | Definições válidas | Definição padrão |
|----------------|---|------------------|
| NUMERIC_STRING | <p>'Y' Sim, esta coluna contém cadeias com caracteres numéricos '0', '1', '2', '9'. Ela não contém espaços em branco. IMPORTANTE: Se esta coluna contiver apenas cadeias numéricas seguidas por espaços em branco à direita, não é aconselhável especificar 'Y'.</p> <p>'N' Não, esta coluna não é uma coluna de cadeia numérica ou uma coluna de cadeia numérica que contém espaços em branco.</p> | 'N' |

Ao definir NUMERIC_STRING para 'Y' para uma coluna, você está informando ao otimizador que esta coluna não contém espaços em branco que possam interferir na classificação dos dados da coluna. Esta opção é útil quando a seqüência de intercalação de uma origem de dados é diferente da do DB2. As colunas marcadas com esta opção não serão excluídas da avaliação remota por causa de uma seqüência de intercalação diferente.

Tabela 70. Opções de coluna e suas definições (continuação)

| Opção | Definições válidas | Definição padrão |
|----------------------------|---|------------------|
| VARCHAR_NO_TRAILING_BLANKS | <p>Esta opção se aplica às origens de dados com tipos de dados de caracteres de variáveis que não têm o comprimento preenchido por espaços em branco à direita.</p> <p>Algumas origens de dados, como o Oracle, não têm semânticas de comparação de caracteres de preenchimento em branco que retornam os mesmos resultados que a semântica de comparação do DB2 para Linux, UNIX e Windows. Defina esta opção quando você quiser que ela se aplique apenas a uma coluna VARCHAR ou VARCHAR2 específica em um objeto de origem de dados.</p> <p>'Y' Sim, espaços em branco à direita não existem nesta coluna VARCHAR.</p> <p>Esta origem de dados tem semântica de comparação de caracteres de preenchimento em branco diferente da do servidor federado. As operações de comparação de caracteres serão processadas no servidor federado e não sofrerão pushdown para a origem de dados.</p> <p>'N' Não, espaços em branco à direita estão presentes nesta coluna VARCHAR.</p> <p>Esta origem de dados tem semântica de comparação de caracteres de preenchimento em branco similar à semântica do servidor federado. As operações de comparação de caracteres podem sofrer pushdown para a origem de dados para processamento.</p> | 'N' |

Related concepts:

- “Pushdown analysis” no *Federated Systems Guide*

Related tasks:

- “Global optimization” no *Federated Systems Guide*

Apêndice E. Opções de Mapeamentos de Função para Sistemas Federados

O DB2 fornece mapeamentos padrão entre as funções de origem de dados internas existentes e as funções internas do DB2. 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, você deve criar um mapeamento de função entre uma função de origem de dados e uma função complementar no banco de dados federado.

A principal finalidade das opções de mapeamento de função é fornecer informações sobre o custo potencial da execução de uma função de origem de dados na origem de dados. A análise pushdown determina se uma função na origem de dados é capaz de executar uma função em uma consulta. O otimizador de consulta, se estiver efetuando pushdown do processamento da função para a origem de dados, é a alternativa menos dispendiosa.

As informações estatísticas fornecidas na definição de mapeamento de função ajudam o otimizador de consulta a comparar o custo estimado da execução da função de origem de dados com o custo estimado da execução da função do DB2.

Tabela 71. Opções de mapeamento de função e suas definições

| Opção | Definições válidas | Definição padrão |
|-------------------|---|-------------------------|
| DISABLE | Desativar um mapeamento de função padrão. Os valores válidos são 'Y' e 'N'. | 'N' |
| INITIAL_INSTS | Número estimado de instruções processadas na primeira e na última vez em que a função da origem de dados é chamada. | '0' |
| INITIAL_IOS | Número estimado de E/Ss executadas na primeira e na última vez em que a função da origem de dados é chamada. | '0' |
| IOS_PER_ARGBYTE | Número estimado de E/Ss estendidas para cada byte do conjunto de argumentos transmitido à função da origem de dados. | '0' |
| IOS_PER_INVOC | Número estimado de E/Ss por chamada de uma função de origem de dados. | '0' |
| INSTS_PER_ARGBYTE | Número estimado de instruções processadas para cada byte do conjunto de argumentos transmitido à função da origem de dados. | '0' |

Tabela 71. Opções de mapeamento de função e suas definições (continuação)

| Opção | Definições válidas | Definição padrão |
|------------------|--|-------------------------|
| INSTS_PER_INVOC | Número estimado de instruções processadas por chamada da função de origem de dados. | '450' |
| PERCENT_ARGBYTES | Porcentagem média estimada de bytes do argumento de entrada que a função da origem de dados irá realmente ler. | '100' |
| REMOTE_NAME | Nome da função da origem de dados. | local name |

Apêndice F. Tipos de Servidores Válidos em Instruções SQL

Os tipos de servidores indicam que tipo de origem de dados o servidor irá representar. Os tipos de servidores variam conforme o fornecedor, a finalidade e o sistema operacional. Os valores suportados dependem do wrapper que está sendo utilizado.

É necessário especificar um tipo de servidor válido na instrução CREATE SERVER.

Wrapper CTLIB

As origens de dados do Sybase suportadas pelo software cliente Sybase CTLIB.

| Tipo de Servidor | Origem de Dados |
|------------------|-----------------|
| SYBASE | Sybase |

Wrapper DBLIB

As origens de dados do Sybase ou do Microsoft SQL Server suportadas pelo software cliente DBLIB.

| Tipo de Servidor | Origem de Dados |
|------------------|-----------------|
| SYBASE | Sybase |

Wrapper DJXMSSQL3

As origens de dados do Microsoft SQL Server suportadas pelo driver ODBC 3.0 (ou superior).

| Tipo de Servidor | Origem de Dados |
|------------------|----------------------|
| MSSQLSERVER | Microsoft SQL Server |

Wrapper DRDA

Família DB2

Tabela 72. DB2 para Linux, UNIX e Windows

| Tipo de Servidor | Origem de Dados |
|-------------------------|-------------------------------------|
| DB2/UDB | IBM DB2 Universal Database |
| DB2/6000 | IBM DB2 para AIX |
| DB2/AIX | IBM DB2 para AIX |
| DB2/HPUX | IBM DB2 para HP-UX V1.2 |
| DB2/HP | IBM DB2 para HP-UX |
| DB2/NT | IBM DB2 para Windows NT |
| DB2/EEE | IBM DB2 Enterprise-Extended Edition |
| DB2/CS | IBM DB2 para Common Server |
| DB2/SUN | IBM DB2 para Solaris V1 e V1.2 |
| DB2/PE | IBM DB2 para Personal Edition |
| DB2/2 | IBM DB2 para OS/2 |
| DB2/LINUX | IBM DB2 para Linux |
| DB2/PTX | IBM DB2 para NUMA-Q |
| DB2/SCO | IBM DB2 para SCO Unixware |

Tabela 73. DB2 para iSeries (e AS/400)

| Tipo de Servidor | Origem de Dados |
|-------------------------|-------------------------------|
| DB2/400 | IBM DB2 para iSeries e AS/400 |

Tabela 74. DB2 para z/OS e OS/390

| Tipo de Servidor | Origem de Dados |
|-------------------------|------------------------|
| DB2/ZOS | IBM DB2 para z/OS |
| DB2/390 | IBM DB2 para OS/390 |
| DB2/MVS | IBM DB2 para MVS |

Tabela 75. DB2 Server para VM e VSE

| Tipo de Servidor | Origem de Dados |
|-------------------------|------------------------|
| DB2/VM | IBM DB2 para VM |

Tabela 75. DB2 Server para VM e VSE (continuação)

| Tipo de Servidor | Origem de Dados |
|-------------------------|------------------------|
| DB2/VSE | IBM DB2 para VSE |
| SQL/DS | IBM SQL/DS |

Wrapper Informix

As origens de dados do Informix suportadas pelo software Informix Client SDK.

| Tipo de Servidor | Origem de Dados |
|-------------------------|------------------------|
| INFORMIX | Informix |

Wrapper MSSQLODBC3

As origens de dados do Microsoft SQL Server suportadas pelo driver DataDirect Connect ODBC 3.6.

| Tipo de Servidor | Origem de Dados |
|-------------------------|------------------------|
| MSSQLSERVER | Microsoft SQL Server |

Wrapper NET8

As origens de dados do Oracle suportadas pelo software cliente Oracle NET8.

| Tipo de Servidor | Origem de Dados |
|-------------------------|---------------------------------|
| ORACLE | Oracle Versão 8.0. ou posterior |

Wrapper ODBC

As origens de dados do ODBC suportadas pelo driver ODBC 3.x.

| Tipo de Servidor | Origem de Dados |
|-------------------------|------------------------|
| ODBC | ODBC |

Wrapper OLE DB

Os provedores OLE DB compatíveis com o Microsoft OLE DB 2.0 ou posterior.

| Tipo de Servidor | Origem de Dados |
|------------------|--------------------------|
| nenhum requerido | Qualquer provedor OLE DB |

Wrapper SQLNET

As origens de dados do Oracle suportadas pelo software cliente SQL*Net V1 ou V2.

| Tipo de Servidor | Origem de Dados |
|------------------|---------------------------|
| ORACLE | Oracle V7.3. ou posterior |

Wrapper Teradata

As origens de dados do Teradata suportadas pelo software cliente Teradata V2R3 e V2R4.

| Tipo de Servidor | Origem de Dados |
|------------------|-----------------|
| TERADATA | Teradata |

Apêndice G. Mapeamentos de Tipos de Dados de Avanço Padrão

Quando um pseudônimo é criado para um objeto de origem de dados, o DB2 para Linux, UNIX e Windows ocupa o catálogo global com informações sobre a tabela.

Estas informações incluem o tipo de dados *remoto* para cada coluna e o tipo de dados correspondente do DB2 para Linux, UNIX e Windows. O tipo de dados do DB2 para Linux, UNIX e Windows é referido como o tipo de dados *local*.

O banco de dados federado utiliza os mapeamentos de tipo de dados para determinar qual tipo de dados do DB2 para Linux, UNIX e Windows deve ser definido para a coluna de um objeto de origem de dados.

Os tipos de dados da origem de dados devem mapear os tipos de dados correspondentes do DB2 para Linux, UNIX e Windows de forma que o servidor federado possa recuperar dados das origens de dados. Para a maioria das origens de dados, os mapeamentos de tipo padrão estão em wrappers. Os mapeamentos de tipo padrão para as origens de dados da família DB2 estão no wrapper DRDA. Os mapeamentos de tipo padrão para o Informix estão no wrapper INFORMIX e assim por diante.

Os servidores federados do DB2 para Linux, UNIX e Windows não suportam mapeamentos para estes tipos de dados locais:

- DATALINK
- tipos definido pelo usuário

Existem dois tipos de mapeamentos entre tipos de dados de origem de dados e tipos de dados do banco de dados federado: mapeamentos de tipos de avanço e mapeamentos de tipos reversos. Em um *mapeamento de tipo de avanço*, o mapeamento ocorre de um tipo remoto para um tipo local comparável.

Você pode substituir um mapeamento do tipo padrão ou criar um novo tipo de mapeamento com a instrução CREATE TYPE MAPPING.

As tabelas a seguir mostram os mapeamentos de avanço padrão entre os tipos de dados do DB2 para Linux, UNIX e Windows e os tipos de dados de origem de dados.

Estes mapeamentos são válidos com todas as versões suportadas, a menos que indicado de outra forma.

Importante: Para todos os mapeamentos de tipos de dados de avanço padrão de uma origem de dados para o DB2 para Linux, UNIX e Windows, o esquema federado do DB2 é SYSIBM.

Origens de Dados DB2 para z/OS e OS/390

Tabela 76. Mapeamentos de tipo de dados de avanço padrão do DB2 para z/OS e OS/390 (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| 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 | - | 0 | - |

Tabela 76. Mapeamentos de tipo de dados de avanço padrão do DB2 para z/OS e OS/390 (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| TIMESTMP | - | - | - | - | - | - | TIMESTAMP | - | 0 | - |
| 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 |

Origens de Dados DB2 para iSeries

Tabela 77. Mapeamentos de tipo de dados de avanço padrão do DB2 para iSeries (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| 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 |

Tabela 77. Mapeamentos de tipo de dados de avanço padrão do DB2 para iSeries (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| 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 | - |
| NUMERIC | - | - | - | - | - | - | DECIMAL | - | - | - |
| SMALLINT | - | - | - | - | - | - | SMALLINT | - | 0 | - |
| TIME | - | - | - | - | - | - | TIME | - | 0 | - |
| TIMESTAMP | - | - | - | - | - | - | TIMESTAMP | - | 0 | - |
| TIMESTMP | - | - | - | - | - | - | TIMESTAMP | - | 0 | - |
| 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 |

Origens de Dados DB2 Server para VM e VSE

Tabela 78. Mapeamentos de tipo de dados de avanço padrão do DB2 Server para VM e VSE (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| 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 | - | 0 | - |
| TIMESTMP | - | - | - | - | - | - | TIMESTAMP | - | 0 | - |
| VARCHAR | 1 | 32672 | - | - | - | - | VARCHAR | - | 0 | N |
| VARCHAR | 1 | 32672 | - | - | S | - | VARCHAR | - | 0 | S |
| VARGRAPHIC | 1 | 16336 | - | - | - | - | VARGRAPHIC | - | 0 | N |
| VARGRAPH | 1 | 16336 | - | - | - | - | VARGRAPHIC | - | 0 | N |

Origens de Dados DB2 para Linux, UNIX e Windows

Tabela 79. Mapeamentos de tipo de dados de avanço padrão do DB2 para Linux, UNIX e Windows (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| BIGINT | - | - | - | - | - | - | BIGINT | - | 0 | - |
| BLOB | - | - | - | - | - | - | BLOB | - | - | - |
| CHAR | - | - | - | - | - | - | CHAR | - | 0 | N |
| CHAR | - | - | - | - | S | - | CHAR | - | 0 | S |
| CLOB | - | - | - | - | - | - | CLOB | - | - | - |
| DATE | - | - | - | - | - | - | DATE | - | 0 | - |
| DBCLOB | - | - | - | - | - | - | DBCLOB | - | - | - |
| DECIMAL | - | - | - | - | - | - | DECIMAL | - | - | - |
| DOUBLE | - | - | - | - | - | - | DOUBLE | - | - | - |
| 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 | - | - | - | - | - | - | TIMESTAMP | - | 0 | - |
| TIMESTMP | - | - | - | - | - | - | TIMESTAMP | - | 0 | - |
| VARCHAR | - | - | - | - | - | - | VARCHAR | - | 0 | N |
| VARCHAR | - | - | - | - | S | - | VARCHAR | - | 0 | S |

Tabela 79. Mapeamentos de tipo de dados de avanço padrão do DB2 para Linux, UNIX e Windows (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| VARGRAPH | - | - | - | - | - | - | VARGRAPHIC | - | 0 | N |
| VARGRAPHIC | - | - | - | - | - | - | VARGRAPHIC | - | 0 | N |

Origens de Dados Informix

Tabela 80. Mapeamentos de tipo de dados de avanço padrão Informix (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| BLOB | - | - | - | - | - | - | BLOB | 2147483647 | - | - |
| BOOLEANO | - | - | - | - | - | - | CHARACTER | 1 | - | - |
| BYTE | - | - | - | - | - | - | BLOB | 2147483647 | - | - |
| CHAR | 1 | 254 | - | - | - | - | CHARACTER | - | - | - |
| CHAR | 255 | 32672 | - | - | - | - | VARCHAR | - | - | - |
| CLOB | - | - | - | - | - | - | CLOB | 2147483647 | - | - |
| DATE | - | - | - | - | - | - | DATE | 4 | - | - |
| DATETIME | 0 | 4 | 0 | 4 | - | - | DATE | 4 | - | - |
| DATETIME | 6 | 10 | 6 | 10 | - | - | TIME | 3 | - | - |
| DATETIME | 0 | 4 | 6 | 15 | - | - | TIMESTAMP | 10 | - | - |
| DATETIME | 6 | 10 | 11 | 15 | - | - | TIMESTAMP | 10 | - | - |
| DECIMAL | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| DECIMAL | 32 | 32 | - | - | - | - | DOUBLE | 8 | - | - |
| 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 | - | - | - |

Tabela 80. Mapeamentos de tipo de dados de avanço padrão Informix (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| 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:

- Para o tipo de dados Informix DATETIME, o servidor federado DB2 UNIX e Windows utiliza o qualificador de alto nível Informix como REMOTE_LENGTH e o qualificador de baixo nível Informix como REMOTE_SCALE.

Os qualificadores Informix são as constantes "TU_" definidas no arquivo datatime.h do Informix Client SDK. 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) | |

Origens de Dados Microsoft SQL Server

Tabela 81. Mapeamentos de tipo de dados de avanço padrão do Microsoft SQL Server (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|------------------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| 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 | 10 | - | - |
| datetimen | - | - | - | - | - | - | TIMESTAMP | 10 | - | - |
| decimal | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| decimal | 32 | 38 | 0 | 38 | - | - | DOUBLE | - | - | - |
| decimaln | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| decimaln | 32 | 38 | 0 | 38 | - | - | DOUBLE | - | - | - |
| DUMMY65 ¹ | 1 | 38 | -84 | 127 | - | - | DOUBLE | - | - | - |
| DUMMY2000 ³ | 1 | 38 | -84 | 127 | - | - | DOUBLE | - | - | - |
| float | - | 8 | - | - | - | - | DOUBLE | 8 | - | - |
| floatn | - | 8 | - | - | - | - | DOUBLE | 8 | - | - |
| float | - | 4 | - | - | - | - | REAL | 4 | - | - |
| floatn | - | 4 | - | - | - | - | REAL | 4 | - | - |
| image | - | - | - | - | - | - | BLOB | 2147483647 | - | S |
| int | - | - | - | - | - | - | INTEGER | 4 | - | - |
| intn | - | - | - | - | - | - | INTEGER | 4 | - | - |
| money | - | - | - | - | - | - | DECIMAL | 19 | 4 | - |
| moneyn | - | - | - | - | - | - | DECIMAL | 19 | 4 | - |

Tabela 81. Mapeamentos de tipo de dados de avanço padrão do Microsoft SQL Server (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|--------------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| nchar | 1 | 127 | - | - | - | - | CHAR | - | - | N |
| nchar | 128 | 4000 | - | - | - | - | VARCHAR | - | - | N |
| numérico | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| numérico | 32 | 38 | 0 | 38 | - | - | DOUBLE | 8 | - | - |
| numericn | 32 | 38 | 0 | 38 | - | - | DOUBLE | - | - | - |
| numericn | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| ntext ² | - | - | - | - | - | - | CLOB | 2147483647 | - | S |
| nvarchar | 1 | 4000 | - | - | - | - | VARCHAR | - | - | N |
| real | - | - | - | - | - | - | REAL | 4 | - | - |
| smallint | - | - | - | - | - | - | SMALLINT | 2 | - | - |
| smalldatetime | - | - | - | - | - | - | TIMESTAMP | 10 | - | - |
| smallmoney | - | - | - | - | - | - | DECIMAL | 10 | 4 | - |
| smallmoneyn | - | - | - | - | - | - | DECIMAL | 10 | 4 | - |
| SQL_BIGINT | - | - | - | - | - | - | DECIMAL | - | - | - |
| SQL_BINARY | 1 | 254 | - | - | - | - | CHARACTER | - | - | S |
| SQL_BINARY | 255 | 8000 | - | - | - | - | VARCHAR | - | - | S |
| SQL_BIT | - | - | - | - | - | - | SMALLINT | 2 | - | - |
| 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_DECIMAL | 32 | 32 | 0 | 31 | - | - | DOUBLE | 8 | - | - |
| SQL_DOUBLE | - | - | - | - | - | - | DOUBLE | 8 | - | - |

Tabela 81. Mapeamentos de tipo de dados de avanço padrão do Microsoft SQL Server (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-------------------------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| SQL_FLOAT | - | - | - | - | - | - | DOUBLE | 8 | - | - |
| SQL_GUID ² | 1 | 4000 | - | - | S | - | VARCHAR | 16 | - | S |
| SQL_INTEGER | - | - | - | - | - | - | INTEGER | 4 | - | - |
| SQL_LONGVARCHAR | - | - | - | - | - | - | CLOB | 2147483647 | - | N |
| SQL_LONGVARBINARY | - | - | - | - | - | - | BLOB | - | - | S |
| SQL_NUMERIC | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| SQL_REAL | - | - | - | - | - | - | DOUBLE | 8 | - | - |
| SQL_SMALLINT | - | - | - | - | - | - | SMALLINT | 2 | - | - |
| SQL_TIME | - | - | - | - | - | - | TIME | 3 | - | - |
| SQL_TIMESTAMP | - | - | - | - | - | - | TIMESTAMP | 10 | - | - |
| SQL_TINYINT | - | - | - | - | - | - | SMALLINT | 2 | - | - |
| SQL_VARBINARY | 1 | 8000 | - | - | - | - | VARCHAR | - | - | S |
| SQL_VARCHAR | 1 | 8000 | - | - | - | - | VARCHAR | - | - | N |
| sysname | - | - | - | - | - | - | VARCHAR | 30 | - | S |
| sysname | 1 | 254 | - | - | - | - | CHAR | - | - | N |
| text | - | - | - | - | - | - | CLOB | - | - | N |
| timestamp | - | - | - | - | - | - | VARCHAR | 8 | - | S |
| tinyint | - | - | - | - | - | - | SMALLINT | 2 | - | - |
| uniqueidentifier ² | 1 | 4000 | - | - | S | - | VARCHAR | 16 | - | S |
| varbinary | 1 | 8000 | - | - | - | - | VARCHAR | - | - | S |
| varchar | 1 | 8000 | - | - | - | - | VARCHAR | - | - | N |

Tabela 81. Mapeamentos de tipo de dados de avanço padrão do Microsoft SQL Server (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|

Notas:

1. Este mapeamento de tipo é válido apenas com o Microsoft SQL Server Versão 6.5.
2. Este mapeamento de tipo é válido apenas com o Microsoft SQL Server Versão 7 e Versão 2000.
3. Este mapeamento de tipo é válido apenas com os sistemas operacionais Windows 2000.

Origens de Dados ODBC

Tabela 82. Mapeamentos de tipo de dados de avanço padrão do ODBC (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| SQL_BIGINT | - | - | - | - | - | - | BIGINT | 8 | - | - |
| 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 |

Tabela 82. Mapeamentos de tipo de dados de avanço padrão do ODBC (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|--------------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| SQL_DECIMAL | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| SQL_DECIMAL | 32 | 38 | 0 | 38 | - | - | DOUBLE | 8 | - | - |
| SQL_DOUBLE | - | - | - | - | - | - | DOUBLE | 8 | - | - |
| SQL_FLOAT | - | - | - | - | - | - | DOUBLE | 8 | - | - |
| SQL_INTEGER | - | - | - | - | - | - | INTEGER | 4 | - | - |
| SQL_LONGVARCHAR | - | - | - | - | - | - | CLOB | 2147483647 | - | N |
| SQL_LONGVARBINARY | - | - | - | - | - | - | BLOB | - | - | 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_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 |

Origens de Dados Oracle NET8

Tabela 83. Mapeamentos de tipo de dados de avanço padrão do Oracle NET8 (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| BLOB | 0 | 0 | 0 | 0 | - | \0 | BLOB | 2147483647 | 0 | S |
| CHAR | 1 | 254 | 0 | 0 | - | \0 | CHAR | 0 | 0 | N |
| CHAR | 255 | 32672 | 0 | 0 | - | \0 | VARCHAR | 0 | 0 | N |
| CLOB | 0 | 0 | 0 | 0 | - | \0 | CLOB | 2147483647 | 0 | N |
| DATE | 0 | 0 | 0 | 0 | - | \0 | TIMESTAMP | 0 | 0 | N |
| FLOAT | 1 | 63 | 0 | 0 | - | \0 | REAL | 0 | 0 | N |
| FLOAT | 64 | 126 | 0 | 0 | - | \0 | DOUBLE | 0 | 0 | N |
| MLSLABEL | 0 | 0 | 0 | 0 | - | \0 | VARCHAR | 255 | 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 |
| RAW | 1 | 254 | 0 | 0 | - | \0 | CHAR | 0 | 0 | S |
| RAW | 255 | 32672 | 0 | 0 | - | \0 | VARCHAR | 0 | 0 | S |
| ROWID | 0 | 0 | 0 | NULL | - | \0 | CHAR | 18 | 0 | N |
| VARCHAR2 | 1 | 32672 | 0 | 0 | - | \0 | VARCHAR | 0 | 0 | N |

Origens de Dados Oracle SQLNET

Tabela 84. Mapeamentos de tipo de dados de avanço padrão do Oracle SQLNET (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| CHAR | 1 | 254 | 0 | 0 | - | \0 | CHAR | 0 | 0 | N |
| CHAR | 255 | 32672 | 0 | 0 | - | \0 | VARCHAR | 0 | 0 | N |
| DATE | 0 | 0 | 0 | 0 | - | \0 | TIMESTAMP | 0 | 0 | N |
| FLOAT | 1 | 63 | 0 | 0 | - | \0 | REAL | 0 | 0 | N |
| FLOAT | 64 | 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 |
| MLSLABEL | 0 | 0 | 0 | 0 | - | \0 | VARCHAR | 255 | 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 |
| RAW | 1 | 254 | 0 | 0 | - | \0 | CHAR | 0 | 0 | S |
| RAW | 255 | 32672 | 0 | 0 | - | \0 | VARCHAR | 0 | 0 | S |
| ROWID | 0 | 0 | 0 | NULL | - | \0 | CHAR | 18 | 0 | N |
| VARCHAR2 | 1 | 32672 | 0 | 0 | - | \0 | VARCHAR | 0 | 0 | N |

Origens de Dados Sybase

Tabela 85. Mapeamentos de tipo de dados de avanço padrão do Sybase CTLIB e DBLIB (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| binary | 1 | 254 | - | - | - | - | CHAR | - | - | S |
| binary | 255 | 16384 | - | - | - | - | VARCHAR | - | - | S |
| bit | - | - | - | - | - | - | SMALLINT | - | - | - |
| char | 1 | 254 | - | - | - | - | CHAR | - | - | N |
| char | 255 | 16384 | - | - | - | - | VARCHAR | - | - | N |
| datetime | - | - | - | - | - | - | TIMESTAMP | - | - | - |
| 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 | 16384 | - | - | - | - | VARCHAR | - | - | N |

Tabela 85. Mapeamentos de tipo de dados de avanço padrão do Sybase CTLIB e DBLIB (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| numérico | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| numérico | 32 | 38 | 0 | 38 | - | - | DOUBLE | - | - | - |
| numericn | 1 | 31 | 0 | 31 | - | - | DECIMAL | - | - | - |
| numericn | 32 | 38 | 0 | 38 | - | - | DOUBLE | - | - | - |
| nvarchar | 1 | 16384 | - | - | - | - | VARCHAR | - | - | N |
| real | - | - | - | - | - | - | REAL | - | - | - |
| smalldatetime | - | - | - | - | - | - | TIMESTAMP | - | - | - |
| smallint | - | - | - | - | - | - | SMALLINT | - | - | - |
| smallmoney | - | - | - | - | - | - | DECIMAL | 10 | 4 | - |
| sysname | 1 | 254 | - | - | - | - | CHAR | - | - | N |
| text | - | - | - | - | - | - | CLOB | - | - | - |
| timestamp | - | - | - | - | - | - | VARCHAR | 8 | - | S |
| tinyint | - | - | - | - | - | - | SMALLINT | - | - | - |
| varbinary | 1 | 16384 | - | - | - | - | VARCHAR | - | - | S |
| varchar | 1 | 16384 | - | - | - | - | VARCHAR | - | - | N |

Origens de Dados Teradata

Tabela 86. Mapeamentos de tipo de dados de avanço padrão do Teradata (Nem todas as colunas mostradas)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|------------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| 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 | - | - | - |
| DATE | - | - | - | - | - | - | DATE | - | - | - |
| DECIMAL | 1 | 18 | 0 | 18 | - | - | 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 | - | - | - |
| TIMESTAMP | - | - | - | - | - | - | TIMESTAMP | - | - | - |
| VARBYTE | 1 | 32762 | - | - | - | - | VARCHAR | - | - | S |
| VARBYTE | 32763 | 64000 | - | - | - | - | BLOB | - | - | - |

Tabela 86. Mapeamentos de tipo de dados de avanço padrão do Teradata (Nem todas as colunas mostradas) (continuação)

| REMOTE_TYPENAME | REMOTE_LOWER_LEN | REMOTE_UPPER_LEN | REMOTE_LOWER_SCALE | REMOTE_UPPER_SCALE | REMOTE_BIT_DATA | REMOTE_DATA_OPERATORS | FEDERATED_TYPENAME | FEDERATED_LENGTH | FEDERATED_SCALE | FEDERATED_BIT_DATA |
|-----------------|------------------|------------------|--------------------|--------------------|-----------------|-----------------------|--------------------|------------------|-----------------|--------------------|
| VARCHAR | 1 | 32672 | - | - | - | - | VARCHAR | - | - | - |
| VARCHAR | 32673 | 64000 | - | - | - | - | CLOB | - | - | - |
| VARGRAPHIC | 1 | 16336 | - | - | - | - | VARGRAPHIC | - | - | - |
| VARGRAPHIC | 16337 | 32000 | - | - | - | - | DBCLOB | - | - | - |

Related concepts:

- “Avançar e Reverter Mapeamentos de Tipos de Dados” no *Federated Systems Guide*

Apêndice H. Mapeamentos de Tipos de Dados Reversos Padrão

Existem dois tipos de mapeamentos entre tipos de dados de origem de dados e tipos de dados do banco de dados federado: mapeamentos de tipos de avanço e mapeamentos de tipos reversos. Em um *mapeamento de tipo de avanço*, o mapeamento ocorre de um tipo remoto para um tipo local comparável. O outro tipo de mapeamento é um *mapeamento de tipo reverso*, que é utilizado com DDL transparente para criar ou modificar tabelas remotas.

Para a maioria das origens de dados, os mapeamentos de tipo padrão estão em wrappers. Os mapeamentos de tipo padrão para as origens de dados da família DB2 estão no wrapper DRDA. Os mapeamentos de tipo padrão para o Informix estão no wrapper INFORMIX e assim por diante.

Ao definir uma tabela ou exibição remota para o banco de dados federado DB2, a definição inclui um mapeamento de tipo reverso. O mapeamento é de um tipo de dados *local* do DB2 para Linux, UNIX e Windows para cada coluna e o tipo de dados *remoto* correspondente. Por exemplo, há um mapeamento de tipo reverso padrão no qual o tipo local REAL aponta para o tipo Informix SMALLFLOAT.

Os servidor federados do DB2 para Linux, UNIX e Windows não suportam mapeamentos para estes tipos de dados locais: LONG VARCHAR, LONG VARGRAPHIC, DATALINK e 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. Estes mapeamentos de tipo reverso padrão atribuirão tipos remotos correspondentes a essas colunas. Por exemplo, suponhamos que você utilize a instrução CREATE TABLE para definir uma tabela Informix com uma coluna C2. Você especifica BIGINT como o tipo de dados para C2 na instrução. O mapeamento de tipo reverso padrão de BIGINT depende de qual versão do Informix está sendo utilizada para criar a tabela. O mapeamento para C2 na tabela do Informix será DECIMAL no Informix Versão 7 e INT8 no Informix Versão 8.

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 DB2 para Linux, UNIX e Windows e os tipos de dados de origem de dados remotos.

Estes mapeamentos são válidos com todas as versões suportadas, a menos que indicado de outra forma.

Origens de Dados do DB2 para z/OS e OS/390

Tabela 87. Mapeamentos de tipo de dados reverso padrão do DB2 para z/OS e OS/390 (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| 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 | - | 10 | - | - | - | - | TIMESTAMP | - | - | - |
| VARCHAR | - | - | - | - | - | - | VARCHAR | - | - | N |
| VARCHAR | - | - | - | - | S | - | VARCHAR | - | - | S |
| VARGRAPHIC | - | - | - | - | - | - | VARGRAPHIC | - | - | N |

Origens de Dados do DB2 para iSeries

Tabela 88. Mapeamentos de tipo de dados reverso padrão do DB2 para iSeries (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| 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 | - | 10 | - | - | - | - | TIMESTAMP | - | - | - |
| VARCHAR | - | - | - | - | - | - | VARCHAR | - | - | N |
| VARCHAR | - | - | - | - | S | - | VARCHAR | - | - | S |
| VARGRAPHIC | - | - | - | - | - | - | VARG | - | - | N |

Origens de Dados do DB2 para VM e VSE

Tabela 89. Mapeamentos de tipo de dados reverso padrão do DB2 para VM e VSE (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| 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 | - | 10 | - | - | - | - | TIMESTAMP | - | - | - |
| VARCHAR | - | - | - | - | - | - | VARCHAR | - | - | - |
| VARCHAR | - | - | - | - | S | - | VARCHAR | - | - | S |
| VARGRAPH | - | - | - | - | - | - | VARGRAPH | - | - | N |

Origens de Dados do DB2 para Linux, UNIX e Windows

Tabela 90. Mapeamentos de tipo de dados reverso padrão do DB2 para Linux, UNIX e Windows (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | FEDERATED_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|--------------------|
| BIGINT | - | 8 | - | - | - | - | BIGINT | - | - | - |
| 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 | - | - | - | - | - | - | REAL | - | - | - |
| SMALLINT | - | 2 | - | - | - | - | SMALLINT | - | - | - |
| TIME | - | 3 | - | - | - | - | TIME | - | - | - |
| TIMESTAMP | - | 10 | - | - | - | - | TIMESTAMP | - | - | - |
| VARCHAR | - | - | - | - | - | - | VARCHAR | - | - | N |
| VARCHAR | - | - | - | - | S | - | VARCHAR | - | - | S |
| VARGRAPH | - | - | - | - | - | - | VARGRAPHIC | - | - | N |
| VARGRAPHIC | - | - | - | - | - | - | VARGRAPHIC | - | - | - |

Origens de Dados do Informix

Tabela 91. Mapeamentos de tipo de dados reverso padrão do Informix (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|----------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| BIGINT ¹ | - | - | - | - | - | - | DECIMAL | 19 | - | - |
| 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 | - | - | - |
| LONG VARCHAR | - | 32700 | - | - | N | - | TEXT | - | - | - |
| LONG VARCHAR | - | 32700 | - | - | S | - | BYTE | - | - | - |
| REAL | - | 4 | - | - | - | - | SMALLFLOAT | - | - | - |
| SMALLINT | - | 2 | - | - | - | - | INTEGER | - | - | - |
| TIME | - | 3 | - | - | - | - | DATETIME | 6 | 10 | - |
| TIMESTAMP | - | 10 | - | - | - | - | DATETIME | 0 | 15 | - |
| VARCHAR | 1 | 254 | - | - | N | - | VARCHAR | - | - | - |
| VARCHAR | 255 | 32672 | - | - | N | - | TEXT | - | - | - |
| VARCHAR | - | - | - | - | S | - | BYTE | - | - | - |
| VARCHAR ² | 255 | 2048 | - | - | N | - | LVARCHAR | - | - | - |
| VARCHAR ² | 2049 | 32672 | - | - | N | - | TEXT | - | - | - |

Tabela 91. Mapeamentos de tipo de dados reverso padrão do Informix (Nem todas as colunas mostradas) (continuação)

| | | | | | | | | | | |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|

Notas:

1. Este mapeamento de tipo é válido apenas com o servidor Informix Versão 7 (ou inferior).
2. Este mapeamento de tipo é válido apenas com o servidor Informix Versão 8 (ou superior).

Para o tipo de dados Informix DATETIME, o servidor federado DB2 UNIX e Windows utiliza o qualificador de alto nível Informix como REMOTE_LENGTH e o qualificador de baixo nível Informix como REMOTE_SCALE.

Os qualificadores Informix são as constantes "TU_" definidas no arquivo `datatime.h` do Informix Client SDK. 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) | |

Origens de Dados Microsoft SQL Server

Tabela 92. Mapeamentos de tipo de dados reverso padrão do Microsoft SQL Server (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| BLOB | - | - | - | - | - | - | image | - | - | - |
| CHARACTER | - | - | - | - | S | - | binary | - | - | - |
| CHARACTER | - | - | - | - | N | - | char | - | - | - |
| CLOB | - | - | - | - | - | - | text | - | - | - |
| DATE | - | 4 | - | - | - | - | data e hora | - | - | - |
| DECIMAL | - | - | - | - | - | - | decimal | - | - | - |
| DOUBLE | - | 8 | - | - | - | - | float | - | - | - |
| INTEGER | - | - | - | - | - | - | int | - | - | - |
| SMALLINT | - | - | - | - | - | - | smallint | - | - | - |
| REAL | - | 4 | - | - | - | - | real | - | - | - |
| TIME | - | 3 | - | - | - | - | data e hora | - | - | - |
| TIMESTAMP | - | 10 | - | - | - | - | data e hora | - | - | - |
| VARCHAR | 1 | 8000 | - | - | N | - | varchar | - | - | - |
| VARCHAR | 8001 | 32672 | - | - | N | - | text | - | - | - |
| VARCHAR | 1 | 8000 | - | - | S | - | varbinary | - | - | - |
| VARCHAR | 8001 | 32672 | - | - | S | - | image | - | - | - |

Origens de Dados do Oracle SQLNET

Tabela 93. Mapeamentos de tipo de dados reverso padrão do Oracle SQLNET (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| BLOB | 0 | 2147483647 | 0 | 0 | S | \0 | LONG RAW | 0 | 0 | S |
| CHARACTER | 1 | 254 | 0 | 0 | N | \0 | CHAR | 0 | 0 | N |
| CHARACTER | 0 | 0 | 0 | 0 | S | \0 | RAW | 0 | 0 | S |
| CLOB | 0 | 2147483647 | 0 | 0 | N | \0 | LONG | 0 | 0 | N |
| DATE | 0 | 4 | 0 | 0 | N | \0 | DATE | 0 | 0 | N |
| DECIMAL | 0 | 0 | 0 | 0 | N | \0 | NUMBER | 0 | 0 | N |
| DOUBLE | 0 | 8 | 0 | 0 | N | \0 | FLOAT | 126 | 0 | N |
| INTEGER | 0 | 4 | 0 | 0 | N | \0 | NUMBER | 9 | 0 | N |
| REAL | 0 | 4 | 0 | 0 | N | \0 | FLOAT | 63 | 0 | N |
| SMALLINT | 0 | 2 | 0 | 0 | N | \0 | NUMBER | 4 | 0 | N |
| TIME | 0 | 3 | 0 | 0 | N | \0 | DATE | 0 | 0 | N |
| TIMESTAMP | 0 | 10 | 0 | 0 | N | \0 | DATE | 0 | 0 | N |
| VARCHAR | 1 | 2000 | 0 | 0 | S | \0 | RAW | 0 | 0 | S |
| VARCHAR | 1 | 4000 | 0 | 0 | N | \0 | VARCHAR2 | 0 | 0 | N |

Nota: o tipo de dados BIGINT do DB2 para Linux, UNIX e Windows não está disponível para DDL transparente. Você não pode especificar o tipo de dados BIGINT em uma instrução CREATE TABLE ao criar uma tabela Oracle remota.

Origens de Dados do Oracle NET8

Tabela 94. Mapeamentos de tipo de dados reverso padrão do Oracle NET8 (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| BLOB | 0 | 2147483647 | 0 | 0 | S | \0 | BLOB | 0 | 0 | S |
| CHARACTER | 1 | 254 | 0 | 0 | N | \0 | CHAR | 0 | 0 | N |
| CHARACTER | 0 | 0 | 0 | 0 | S | \0 | RAW | 0 | 0 | S |
| CLOB | 0 | 2147483647 | 0 | 0 | N | \0 | CLOB | 0 | 0 | N |
| DATE | 0 | 4 | 0 | 0 | N | \0 | DATE | 0 | 0 | N |
| DECIMAL | 0 | 0 | 0 | 0 | N | \0 | NUMBER | 0 | 0 | N |
| DOUBLE | 0 | 8 | 0 | 0 | N | \0 | FLOAT | 126 | 0 | N |
| INTEGER | 0 | 4 | 0 | 0 | N | \0 | NUMBER | 9 | 0 | N |
| REAL | 0 | 4 | 0 | 0 | N | \0 | FLOAT | 63 | 0 | N |
| SMALLINT | 0 | 2 | 0 | 0 | N | \0 | NUMBER | 4 | 0 | N |
| TIME | 0 | 3 | 0 | 0 | N | \0 | DATE | 0 | 0 | N |
| TIMESTAMP | 0 | 10 | 0 | 0 | N | \0 | DATE | 0 | 0 | N |
| VARCHAR | 1 | 4000 | 0 | 0 | N | \0 | VARCHAR2 | 0 | 0 | N |
| VARCHAR | 1 | 2000 | 0 | 0 | S | \0 | RAW | 0 | 0 | S |

Nota: O tipo de dados BIGINT do DB2 para Linux, UNIX e Windows não está disponível para DDL transparente. Você não pode especificar o tipo de dados BIGINT em uma instrução CREATE TABLE ao criar uma tabela Oracle remota.

Origens de Dados do Sybase

Tabela 95. Mapeamentos de tipo de dados reverso padrão do Sybase CTLIB e DBLIB (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | REMOTE_BIT_DATA |
|--------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|-----------------|
| BIGINT | - | - | - | - | - | - | decimal | 19 | 0 | - |
| BLOB | - | - | - | - | - | - | image | - | - | - |
| CHARACTER | - | - | - | - | N | - | char | - | - | - |
| CHARACTER | - | - | - | - | S | - | binary | - | - | - |
| CLOB | - | - | - | - | - | - | text | - | - | - |
| DATE | - | - | - | - | - | - | data e hora | - | - | - |
| DECIMAL | - | - | - | - | - | - | decimal | - | - | - |
| DOUBLE | - | - | - | - | - | - | float | - | - | - |
| INTEGER | - | - | - | - | - | - | integer | - | - | - |
| REAL | - | - | - | - | - | - | real | - | - | - |
| SMALLINT | - | - | - | - | - | - | smallint | - | - | - |
| TIME | - | - | - | - | - | - | data e hora | - | - | - |
| TIMESTAMP | - | - | - | - | - | - | data e hora | - | - | - |
| VARCHAR | 1 | 255 | - | - | N | - | varchar | - | - | - |
| VARCHAR | 256 | 32672 | - | - | N | - | text | - | - | - |
| VARCHAR | 1 | 255 | - | - | S | - | varbinary | - | - | - |
| VARCHAR | 256 | 32672 | - | - | S | - | image | - | - | - |

Origens de Dados do Teradata

Tabela 96. Mapeamentos de tipo de dados reverso padrão do Teradata (Nem todas as colunas mostradas)

| FEDERATED_TYPENAME | FEDERATED_LOWER_LEN | FEDERATED_UPPER_LEN | FEDERATED_LOWER_SCALE | FEDERATED_UPPER_SCALE | FEDERATED_BIT_DATA | FEDERATED_DATA_OPERATORS | REMOTE_TYPENAME | REMOTE_LENGTH | REMOTE_SCALE | FEDERATED_BIT_DATA |
|---------------------|---------------------|---------------------|-----------------------|-----------------------|--------------------|--------------------------|-----------------|---------------|--------------|--------------------|
| BLOB ¹ | 1 | 64000 | - | - | - | - | VARBYTE | - | - | - |
| CHARACTER | - | - | - | - | - | - | CHARACTER | - | - | - |
| CHARACTER | - | - | - | - | S | - | BYTE | - | - | - |
| CLOB ² | 1 | 64000 | - | - | - | - | VARCHAR | - | - | - |
| DATE | - | - | - | - | - | - | DATE | - | - | - |
| DBCLOB ³ | 1 | 32000 | - | - | - | - | VARGRAPHIC | - | - | - |
| DECIMAL | 1 | 18 | 0 | 18 | - | - | DECIMAL | - | - | - |
| DECIMAL | 19 | 31 | 0 | 31 | - | - | FLOAT | - | - | - |
| DOUBLE | - | - | - | - | - | - | FLOAT | - | - | - |
| GRAPHIC | - | - | - | - | - | - | GRAPHIC | - | - | - |
| INTEGER | - | - | - | - | - | - | INTEGER | - | - | - |
| REAL | - | - | - | - | - | - | FLOAT | - | - | - |
| SMALLINT | - | - | - | - | - | - | SMALLINT | - | - | - |
| TIME | - | - | - | - | - | - | TIME | - | - | - |
| TIMESTAMP | - | - | - | - | - | - | TIMESTAMP | - | - | - |
| VARCHAR | - | - | - | - | - | - | VARCHAR | - | - | - |
| VARCHAR | - | - | - | - | S | - | VARBYTE | - | - | - |
| VARGRAPHIC | - | - | - | - | - | - | VARGRAPHIC | - | - | - |

Tabela 96. Mapeamentos de tipo de dados reverso padrão do Teradata (Nem todas as colunas mostradas) (continuação)

| |
|--------------------------|
| FEDERATED_TYPENAME |
| FEDERATED_LOWER_LEN |
| FEDERATED_UPPER_LEN |
| FEDERATED_LOWER_SCALE |
| FEDERATED_UPPER_SCALE |
| FEDERATED_BIT_DATA |
| FEDERATED_DATA_OPERATORS |
| REMOTE_TYPENAME |
| REMOTE_LENGTH |
| REMOTE_SCALE |
| FEDERATED_BIT_DATA |

Notas:

1. O tipo de dados Teradata VARBYTE pode conter apenas o comprimento especificado (1 a 64 000) de um tipo de dados DB2 BLOB.
 2. O tipo de dados Teradata VARCHAR pode conter apenas o comprimento especificado (1 a 64 000) de um tipo de dados DB2 CLOB.
 3. O tipo de dados Teradata VARGRAPHIC pode conter apenas o comprimento especificado (1 a 32 000) de um tipo de dados DB2 DBCLOB.
-

Related concepts:

- “Avançar e Reverter Mapeamentos de Tipos de Dados” no *Federated Systems Guide*

Documentação Técnica do DB2 Information Integrator

Os tópicos a seguir descrevem como:

- Acessar manuais e informações sobre release, incluindo impressão e pedido de manuais
- Acessar tópicos utilizando o DB2 Information Integrator Information Center ou o CD de Documentação em HTML do DB2

Acessando Manuais e Informações sobre Release

As informações técnicas sobre o DB2 Information Integrator estão disponíveis nos seguintes formatos:

- Manuais (em PDF e impressos). Uma descrição de cada um dos manuais da biblioteca do DB2 Information Integrator está disponível no IBM Publications Center no endereço www.ibm.com/shop/publications/order.
- Um centro de informações (formato HTML).
- Ajuda para ferramentas de banco dados do DB2 (formato HTML).

Manuais do DB2 Information Integrator

O CD de Documentação em PDF do DB2 Information Integrator contém arquivos em PDF dos manuais da biblioteca do DB2 Information Integrator e da biblioteca do DB2 Universal Database. A estrutura do CD de Documentação em PDF do DB2 Information Integrator está:

- Em sistemas operacionais Windows: *x:\doc\%L*
- Em sistemas operacionais UNIX: */cdrom/doc/%L/*

em que:

- *x* representa a letra da unidade de CD-ROM no Windows
- *cdrom* refere-se ao ponto de montagem do CD-ROM no UNIX
- *%L* é o locale da documentação que você deseja utilizar, por exemplo, *en_US*

| Idioma | Locale | Identific. | Idioma | Locale | Identific. |
|---------------------|--------|------------|---------------------|--------|------------|
| Árabe | ar_AA | w | Japonês | ja_JP | j |
| Português do Brasil | pt_BR | b | Coreano | ko_KR | k |
| Búlgaro | bg_BG | u | Norueguês | no_NO | n |
| Croata | hr_HR | 9 | Polonês | pl_PL | p |
| Tcheco | cs_CZ | x | Português | pt_PT | v |
| Dinamarquês | da_DK | d | Romeno | ro_RO | 8 |
| Holandês | nl_NL | q | Russo | ru_RU | r |
| Inglês | en_US | e | Chinês Simplificado | zh_CN | c |

| | | | | | |
|-----------|-------|---|--------------------|-------|---|
| Finlandês | fi_FI | y | Eslovaco | sk_SK | 7 |
| Francês | fr_FR | f | Esloveno | sl_SI | l |
| Alemão | de_DE | g | Espanhol | es_ES | z |
| Grego | el_GR | a | Sueco | sv_SE | s |
| Húngaro | hu_HU | h | Chinês Tradicional | zh_TW | t |
| Italiano | it_IT | i | Turco | tr_TR | m |

O caractere na sexta posição do nome de cada arquivo PDF indica a versão do idioma de um manual (consulte a tabela a seguir). Por exemplo, o nome de arquivo `iiyige80` identifica a versão em inglês da publicação *IBM DB2 Information Integrator Installation Guide*, e o nome de arquivo `iiyigg80` identifica a versão em alemão do mesmo manual.

Os manuais na tabela a seguir estão disponíveis para o DB2 Information Integrator.

Tabela 97. Documentação do DB2 Information Integrator

| Nome | Número de Formulário | Categoria de Instalação | Nome do Arquivo PDF |
|---|---------------------------|-------------------------|---------------------|
| <i>IBM DB2 Information Integrator Solutions Guide</i> | SC18-7037 | getting_started | iiyisx80 |
| <i>IBM DB2 Information Integrator Installation Guide</i> | GC18-7036 | getting_started | iiyigx80 |
| <i>IBM DB2 Information Integrator Migration Guide</i> | SC18-7360 | getting_started | iiymgx80 |
| <i>IBM DB2 Information Integrator Federated Systems Guide</i> | SC18-7364 | admin | iiyfp80 |
| <i>IBM DB2 Information Integrator Data Source Configuration Guide</i> | Disponível apenas on-line | opcional | iiylsx80 |
| <i>IBM DB2 Information Integrator Developer's Guide</i> | SC18-7359 | ad | iiyfsx80 |

Imprimindo Manuais a partir de Arquivos PDF

Você pode imprimir os manuais do DB2 Information Integrator a partir dos arquivos PDF no CD de Documentação em PDF do DB2 Information Integrator. Você pode utilizar o Adobe Acrobat Reader para imprimir o manual inteiro, um intervalo de páginas ou páginas específicas.

Pré-requisitos:

Certifique-se de que você tenha o Adobe Acrobat Reader. Ele está disponível a partir do Web site da Adobe no endereço www.adobe.com.

Procedimento:

Para imprimir o manual do DB2 Information Integrator a partir de um arquivo PDF:

1. Insira o CD de Documentação em PDF do DB2 Information Integrator. Em sistemas operacionais UNIX, monte o CD.
2. Inicie o Adobe Acrobat Reader.
3. Abra o arquivo PDF a partir de um dos seguintes locais:
 - Em sistemas operacionais Windows: `x:\doc\%L`
 - Em sistemas operacionais UNIX: `/cdrom/doc/%L/`em que:
 - `x` representa a letra da unidade de CD-ROM no Windows
 - `cdrom` refere-se ao ponto de montagem do CD-ROM no UNIX
 - `%L` é o locale da documentação que você deseja imprimir, por exemplo, `en_US`
4. Clique em **File** → **Print**.
5. Na janela Print, especifique se você deseja imprimir todas as páginas, a página atual ou um intervalo de páginas.
6. Clique em **OK**.

Solicitando Manuais Impressos

Você pode obter manuais impressos, solicitando o doc pack (pacote da documentação) para seu produto DB2 Information Integrator ao seu revendedor IBM. Os doc packs são um subconjunto dos manuais da biblioteca do DB2 Information Integrator. Estes doc packs são projetados para ajudá-lo a começar a utilizar o produto DB2 Information Integrator que foi comprado.

Os manuais nos doc packs são os mesmos que aqueles do CD de Documentação em PDF do DB2 Information Integrator, que foi fornecido com seu produto DB2 Information Integrator.

Você também pode utilizar um dos seguintes métodos para pedir manuais individuais:

- Entre em contato com seu representante de marketing ou revendedor autorizado IBM. Para localizar um representante da IBM local, verifique o IBM Worldwide Directory of Contacts no endereço www.ibm.com/planetwide.

- Vá para o IBM Publications Center no endereço www.ibm.com/shop/publications/order.

Notas sobre o Release e Requisitos de Instalação

As notas sobre o release e os requisitos de instalação fornecem informações que são específicas no nível de release e do fix pack de seu produto. As notas sobre o release também fornecem resumos das atualizações da documentação que são incorporadas em cada release e em cada fix pack.

As notas sobre o release e os requisitos de instalação estão disponíveis em formato de texto e em HTML no CD-ROM do produto:

- Em sistemas operacionais Windows: `x:\doc\%L`
- Em sistemas operacionais UNIX: `/cdrom/doc/%L/`

em que:

- `x` representa a letra da unidade de CD-ROM no Windows
- `cdrom` refere-se ao ponto de montagem do CD-ROM no UNIX
- `%L` é o locale da documentação que você deseja utilizar; por exemplo, `en_US`

Tabela 98. Notas sobre o Release

| Nome | Nome de Arquivo | Localização |
|---|-----------------|--|
| <i>DB2 Information Integrator Release Notes</i> | ReleaseNotes | <ul style="list-style-type: none"> • O CD-ROM do produto • O DB2 Information Integrator Information Center • O DB2 Information Integration Installation Launchpad |
| <i>DB2 Information Integrator Installation Requirements</i> | Prereqs | <ul style="list-style-type: none"> • O CD-ROM do produto • O DB2 Information Integration Installation Launchpad |

Fix Packs para a Documentação do DB2 Information Integrator

A IBM pode disponibilizar fix packs da documentação periodicamente. Você pode utilizar os fix packs da documentação para atualizar as informações instaladas a partir do CD de Documentação em HTML do DB2 quando novas informações forem disponibilizadas.

Os fix packs da documentação são cumulativos. Por exemplo, se você instalar a documentação da Versão 8.1 e, em seguida, aplicar a Versão 8.1.2, você obterá as atualizações de documentação para o FixPack 1 e para a Versão 8.1.2.

Quando você instalar os fix packs da documentação, sua documentação em HTML estará mais atualizada do que os manuais impressos ou arquivos PDF on-line de seu produto.

Acessando Tópicos Utilizando o DB2 Information Integrator Information Center ou o CD de Documentação em HTML do DB2

O DB2 Information Integrator Information Center oferece acesso às informações que você precisa para aproveitar as vantagens do DB2 Information Integrator em seus negócios.

Recursos do DB2 Information Integrator Information Center

O DB2 Information Integrator Information Center possui os seguintes recursos:

Árvore de Navegação Integrada

Localiza qualquer tópico na biblioteca do DB2 Information Integrator a partir de uma única árvore de navegação.

Pesquisa

Pesquisa todos os tópicos em sua estação de trabalho, clicando em **Pesquisar** na barra de ferramentas do navegador.

Índice Principal

Acessa as informações em tópicos e ferramentas de ajuda a partir de um índice principal. O índice contém entradas de toda a biblioteca do DB2 Information Integrator.

Glossário Principal

O glossário principal define os termos utilizados na biblioteca do DB2 Information Integrator.

Documentação Regularmente Atualizada

Mantém seu tópicos atualizados, fazendo o download de tópicos de HTML atualizados.

Localizando Tópicos no DB2 Information Integrator Information Center

Os elementos principais a seguir constituem o DB2 Information Integrator Information Center:

Árvore de Navegação

A árvore de navegação está localizada no quadro esquerdo da janela do navegador. A árvore se expande e se reduz para mostrar e ocultar links de tópicos, o glossário e o índice principal no DB2 Information Integrator Information Center.

Barra de Ferramentas de Navegação

A barra de ferramentas de navegação está localizada no quadro direito superior da janela do navegador. Utilize os botões de comando na barra de ferramentas de navegação para pesquisar o DB2 Information

Integrator Information Center, ocultar a árvore de navegação e localizar o tópico atualmente exibido na árvore de navegação.

Quadro de Conteúdo

O quadro de conteúdo está localizado no quadro direito inferior da janela do navegador. Quando você clica em um link na árvore de navegação, clica em um resultado de pesquisa ou segue um link de outro tópico ou do índice principal, o quadro de conteúdo exibe o tópico apropriado.

Pré-requisitos:

Para acessar o DB2 Information Integrator Information Center a partir de um navegador, você deve utilizar os seguintes navegadores:

- Microsoft Explorer Versão 5 ou posterior
- Netscape Navigator Versão 6.1 ou posterior

Restrições:

O DB2 Information Integrator Information Center contém apenas os conjuntos de tópicos que você instalou a partir do CD de Documentação em HTML do DB2. Se seu navegador da Web retornar um erro Arquivo não encontrado quando seguir um link para um tópico, instale um ou mais conjuntos adicionais de tópicos do CD de Documentação em HTML do DB2.

Procedimento:

Para localizar um tópico através de uma pesquisa com palavras-chave:

1. Na barra de ferramentas da navegação, clique em **Pesquisar**.
2. No campo de entrada de texto na parte superior da janela Pesquisar, digite um ou mais termos relacionados à sua área de interesse e, em seguida, clique em **Pesquisar**. O campo **Resultados** exibe uma lista de tópicos. Os tópicos que mais se aproximam de sua cadeia de pesquisa estão na parte superior da lista.
Digite mais termos para aumentar a adequação de sua consulta e para reduzir o número de tópicos retornados para sua consulta.
3. No campo **Resultados**, clique no título do tópico que você deseja ler. O tópico será exibido no quadro de conteúdo.

Para localizar um tópico na árvore de navegação:

1. Na árvore de navegação, clique no ícone do livro próximo à categoria de tópicos nos quais você está interessado. Será exibida uma lista de subcategorias sob o ícone.

2. Continue a clicar nos ícones de livro até encontrar a categoria que contenha os tópicos nos quais você está interessado. Categorias que estabelecem link para tópicos mostram o título da categoria como um link quando você move o cursor sobre o título da categoria. Um ícone da página será utilizado na árvore de navegação para identificar tópicos.
3. Clique no link do tópico. O tópico será exibido no quadro de conteúdo.

Para localizar um tópico utilizando o índice principal:

1. Na árvore de navegação, clique em **Índice**. O índice é expandido para exibir uma lista de links organizados em ordem alfabética.
2. Na árvore de navegação, clique no primeiro caractere do assunto que você está procurando. Uma lista de entradas com este caractere inicial é exibida no quadro de conteúdo. Se for exibido o ícone de um livro, o assunto possui várias entradas de índice.
3. Clique no ícone do livro que corresponde ao assunto no qual você está interessado. É exibida uma lista de tópicos abaixo do termo no qual você clicou.
4. Clique no título do tópico que preenche suas necessidades. O tópico será exibido no quadro de conteúdo.

Utilizando a Documentação em HTML do DB2

Este tópico descreve como instalar, exibir e copiar a documentação no CD de Documentação em HTML do DB2 e como atualizar a documentação após instalá-lo.

Instalando a Documentação em HTML do DB2

O diretório de instalação para o CD de Documentação em HTML do DB2 é diferente para cada categoria de informação:

htmlcdpath/doc/htmlcd/%L/category

htmlcdpath

O diretório em que o CD de Documentação em HTML do DB2 está instalado.

%L O locale da documentação que você deseja utilizar, por exemplo, en_US.

category

O identificador da categoria, por exemplo, getting_started para informações sobre instalação.

Exibindo Documentação Técnica Diretamente do CD de Documentação HTML do DB2

Você pode exibir todos os tópicos em HTML a partir do CD de Documentação em HTML do DB2.

Restrições:

Você deve instalar o produto DB2 para exibir a ajuda on-line.

Procedimento:

Para exibir a documentação em HTML a partir do CD de Documentação em HTML do DB2:

1. Insira o CD de Documentação em HTML do DB2.
2. Inicie seu navegador da Web e abra o arquivo a seguir:
 - Para sistemas operacionais Windows:
x:\Arquivos de programas\IBM\sqliib\doc\htmlcd\%L\index.*
 - Para sistemas operacionais UNIX:
/cdrom/Program Files/IBM/sqliib/doc
/htmlcd/ %L/index.htm

em que:

- *x* representa a letra da unidade de CD-ROM no Windows
- *cdrom* refere-se ao ponto de montagem do CD-ROM no UNIX
- *%L* é o locale da documentação que você deseja utilizar, por exemplo, en_US

Copiando Arquivos do CD de Documentação HTML do DB2 para um Servidor Web

A biblioteca inteira do DB2 está disponível no CD de Documentação em HTML do DB2, portanto, você pode copiar a biblioteca em um servidor Web para obter acesso mais facilmente.

Procedimento:

Copie os arquivos do CD de Documentação em HTML do DB2 no caminho apropriado em seu servidor Web (o caminho padrão é mostrado):

- Para sistemas operacionais Windows: x:\Arquivos de programas\IBM\sqliib\doc\htmlcd\%L*.*
- Para sistemas operacionais UNIX: /cdrom/Program Files/IBM/sqliib/doc/htmlcd/%L

em que:

- *x* representa a letra da unidade de CD-ROM no Windows
- *cdrom* refere-se ao ponto de montagem do CD-ROM no UNIX
- *%L* é o locale da documentação que você deseja utilizar, por exemplo, en_US

Atualizando a Documentação em HTML em seu Computador

Quando a IBM disponibiliza atualizações, você pode atualizar os arquivos HTML que você instalou a partir do CD de Documentação em HTML do DB2:

- Utilizando o Information Center (se você possuir as ferramentas de interface gráfica com o usuário de administração do DB2 instaladas)
- Fazendo o download e aplicando um fix pack da Documentação em HTML do DB2

Este procedimento não atualiza o código do DB2.

Pré-requisitos:

Assegure que seu computador tenha acesso à Internet, pois o atualizador faz downloads do fix pack de documentação mais recente a partir do servidor IBM, se necessário. Para conectar-se à Internet, pode ser necessário fornecer suas informações de proxy.

Procedimento:

Para utilizar o centro de informações para atualizar sua documentação em HTML local:

1. Inicie o DB2 Information Center:
 - A partir das ferramentas de administração gráfica, clique no ícone do **Information Center** na barra de ferramentas.
 - Na linha de comandos, digite db2ic.
2. Clique em **Information Center** → **Atualizar Documentação Local** para iniciar a atualização.

Se uma atualização de documentação estiver disponível, ela será transferida por download e aplicada.

Para fazer download e aplicar a atualização da documentação manualmente:

1. Abra a página de suporte do DB2 em seu navegador da Web no endereço www.ibm.com/software/data/db2/udb/winos2unix/support.
2. Clique em **DB2 Version 8** e localize o link do fix pack da documentação para o seu sistema operacional.
3. Determine se sua documentação do DB2 local está desatualizada, comparando o nível do fix pack com o nível da documentação que você tem instalado.
4. Se uma versão mais recente da documentação estiver disponível, faça o download do fix pack para seu sistema operacional. Há um fix pack para todos os sistemas operacionais Windows e um fix pack para todos os sistemas operacionais UNIX.
5. Aplique o fix pack:
 - Para sistemas operacionais Windows: O fix pack da documentação é um arquivo zip autodescompactável. Faça o download do fix pack da

documentação em qualquer diretório vazio e descompacte-o. Execute o comando **setup** no diretório para instalar o fix pack da documentação.

- Para sistemas operacionais UNIX: O fix pack da documentação é um arquivo tar.Z compactado. Descompacte o arquivo tar para criar um diretório denominado `delta_install`. Execute o script `installdocfix` dentro do diretório para instalar o fix pack da documentação.

Pesquisando a Documentação do DB2

Para pesquisar a documentação do DB2, utilize o Netscape Versão 6.1 (ou posterior) ou o Microsoft Internet Explorer Versão 5 (ou posterior). Assegure que o suporte Java do navegador esteja ativado.

Uma janela de pesquisa é aberta quando você clica no ícone de pesquisa na barra de ferramentas de navegação do DB2 Information Integrator Information Center em um navegador. Se você estiver utilizando a função de pesquisa pela primeira vez, a janela de pesquisa pode demorar algum tempo para carregar.

Restrições:

As seguintes restrições são aplicáveis ao pesquisar a documentação:

- Pesquisas booleanas não são suportadas. Os qualificadores de pesquisa Booleanos *and* e *or* são ignorados em uma pesquisa. Por exemplo, as seguintes pesquisas produzirão os mesmos resultados:
 - `servlets and beans`
 - `servlets or beans`
- As pesquisas com caracteres curingas não são suportadas. Um pesquisa por *java** procurará a cadeia literal *java** e não localizará, por exemplo, *javadoc*.

Em geral, você obterá melhores resultados de pesquisa se pesquisar frases ao invés de palavras simples.

Procedimento:

Para pesquisar a documentação do DB2:

1. Na barra de ferramentas da navegação, clique em **Pesquisar**.
2. No campo de entrada de texto na parte superior da janela Pesquisar, digite um ou mais termos relacionados à sua área de interesse e, em seguida, clique em **Pesquisar**. O campo **Resultados** exibe uma lista de tópicos classificados por adequação.

Digite mais termos para aumentar a adequação de sua consulta e para reduzir o número de tópicos retornados por sua consulta.

3. No campo **Resultados**, clique no título do tópico que você deseja ler. O tópico é exibido no quadro de conteúdo.

Quando você desempenha uma pesquisa, o primeiro resultado é automaticamente carregado em seu no quadro do navegador. Para visualizar o conteúdo de outros resultados de pesquisa, clique no resultado na lista de resultados.

Pesquisando Documentações do DB2 sobre Resolução de Problemas com o Netscape 4.x

A maioria dos problemas é relacionada ao suporte Java fornecido pelos navegadores da Web. Este tópico fornece possíveis soluções.

Procedimento:

Um problema comum com o Netscape 4.x envolve uma ausência ou má colocação da classe de segurança. Tente a solução a seguir, especialmente se você visualizar a seguinte linha no console Java do navegador:

```
Cannot find class java/security/InvalidParameterException
```

Copie o seguinte arquivo do CD de Documentação em HTML do DB2 para o diretório `java\classes\java\security\` dentro do diretório em que o navegador Netscape está instalado. É possível que você precise criar a estrutura do subdiretório `java\security\`.

- Nos sistemas operacionais Windows:

```
x:Arquivos de programas\IBM\sqllib\doc\htmlcd\%L\InvalidParameterException.class
```

- Nos sistemas operacionais UNIX:

```
/cdrom/Program Files/IBM/sqllib/doc/htmlcd/%L  
/InvalidParameterException.class
```

em que:

- *x* representa a letra da unidade de CD-ROM no Windows
- *cdrom* refere-se ao ponto de montagem do CD-ROM no UNIX
- *%L* é o locale da documentação que você deseja utilizar, por exemplo, `en_US`

Se o seu navegador Netscape ainda falhar ao exibir a janela de entrada de pesquisa, tente as seguintes ações:

- Pare todas as instâncias dos navegadores Netscape para assegurar que não haja código de navegação do Netscape em execução no computador. Abra uma nova instância do navegador Netscape e inicie a pesquisa novamente.
- Limpe o cache do navegador.
- Tente uma versão diferente do Netscape ou um navegador diferente.

Acessibilidade

Usuários com deficiências físicas, como mobilidade restrita ou visão limitada, podem utilizar produtos de software de forma bem-sucedida através dos recursos de acessibilidade. Estes são os principais recursos de acessibilidade no DB2 Information Integrator Versão 8:

- Todos os recursos podem ser operados utilizando o teclado em vez do mouse.
- É possível personalizar o tamanho e a cor de suas fontes.
- É possível receber informações visuais ou auditivas sobre alertas.
- O DB2 suporta aplicativos de acessibilidade que utilizam o Java™ Accessibility API.
- A documentação do DB2 é fornecida em um formato acessível.

Entrada de Dados e Navegação através do Teclado

Você pode operar as ferramentas de banco de dados do DB2, tais como Control Center, Data Warehouse Center e Replication Center, utilizando apenas o teclado. É possível utilizar teclas ou combinações de teclas em vez de um mouse para executar a maioria das operações.

Nos sistemas baseados em UNIX, a posição do foco do teclado é realçada. Este realce indica qual área da janela está ativa e onde sua digitação surtirá efeito.

Exibição Acessível

As ferramentas de banco de dados do DB2 possuem recursos que otimizam a interface com o usuário e aprimoram a acessibilidade para usuários com deficiência visual. Estes aperfeiçoamentos de acessibilidade incluem suporte para propriedades de fontes personalizáveis.

Definições das Fontes

Para as ferramentas do banco de dados DB2, você pode utilizar o bloco de notas Definições de Ferramentas para selecionar a cor, o tamanho e a fonte para o texto nos menus e janelas.

Não-dependência de Cores

Não é necessário distinguir cores para utilizar as funções deste produto.

Dicas de Alertas Alternativos

Você pode especificar se quer receber sugestões sobre alertas visuais ou de áudio, utilizando o bloco de notas Definições de Ferramentas.

Compatibilidade com Tecnologias Assistidas

A interface gráfica com o DB2 Information Integrator suporta a Java Accessibility API, ativando a utilização de leitoras de tela e outras tecnologias assistidas que são utilizadas por pessoas com deficiências.

Documentação Acessível

A documentação para os produtos da família DB2 está disponível no formato HTML. Você pode exibir a documentação de acordo com as preferências de exibição definidas em seu navegador. Você pode utilizar leitoras de tela e outras tecnologias assistidas.

Avisos

Estas informações foram desenvolvidas para produtos e serviços oferecidos nos Estados Unidos. É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos neste documento em outros países. Consulte um representante IBM local para obter informações sobre os 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 quaisquer direitos de propriedade intelectual da IBM, poderá ser utilizado em substituição a este produto, programa ou serviço. Entretanto a avaliação e verificação da operação de qualquer produto, programa ou serviço não-IBM são de responsabilidade do Cliente.

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

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

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

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

O parágrafo a seguir não se aplica a nenhum país em que tais disposições não estejam de acordo com a legislação local: A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO “NO ESTADO EM QUE SE ENCONTRA”, SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS NÃO SE LIMITANDO ÀS GARANTIAS IMPLÍCITAS DE NÃO-VIOLAÇÃO, MERCADO OU ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns

países não permitem a exclusão de garantias expressas ou implícitas em certas transações; portanto, esta disposição pode não se aplicar ao Cliente.

Esta publicação pode conter imprecisões técnicas ou erros tipográficos. Periodicamente, são feitas alterações nas informações aqui contidas; tais alterações serão incorporadas em futuras edições desta publicação. A IBM pode, a qualquer momento, 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. Os materiais contidos nesses Web sites não fazem parte dos materiais deste produto IBM e a utilização desses Web sites é de inteira responsabilidade do Cliente.

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

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

Gerência de Relações Comerciais e Industriais da IBM Brasil
Av. Pasteur 138/146
Botafogo
Rio de Janeiro
Cep: 22290-240

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

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

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

As informações referentes a produtos não-IBM foram obtidas junto aos fornecedores desses produtos, anúncios publicados ou outras fontes de publicidade disponíveis. A IBM não testou estes produtos e não pode confirmar a precisão de do desempenho, da compatibilidade ou de qualquer outra reivindicação relacionada a produtos não-IBM. Dúvidas sobre a capacidade de produtos não-IBM devem ser encaminhadas diretamente a seus fornecedores.

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

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

LICENÇA DE COPYRIGHT:

Estas informações podem conter programas aplicativos de exemplo em seu idioma, para ilustrar técnicas de programação em várias plataformas operacionais. Você pode copiar, modificar e distribuir estes programas de exemplo sem a necessidade de pagar a 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. Estes exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar a confiabilidade, manutenção ou função destes programas.

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 de Programas de Exemplo da IBM Corp. © Copyright IBM Corp. *_digite o ano ou anos_*. Todos os direitos reservados.

Marcas Comerciais

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

AIX
DB2
Domino

IBM
Informix
Lotus
Lotus Notes
QuickPlace
WebSphere

Os termos a seguir são marcas comerciais ou marcas de serviço de outras empresas:

Microsoft, Windows, Windows NT e o logotipo Windows são marcas comerciais da Microsoft Corporation nos Estados Unidos e/ou em outros países.

UNIX é uma marca registrada da The Open Group nos Estados Unidos e em outros países.

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

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

Índice Remissivo

A

ajuste

- configuração da origem de dados
 - Informix 60
 - Microsoft SQL Server 108
 - ODBC 124
 - Oracle 76
 - Origens de Dados da Família
 - DB2 44
 - Sybase 92
 - Teradata 141

amostra

- HMMER
 - consultas 349

amostras

- consultas
 - BLAST 232, 240, 241
 - Documentum 188
 - Excel 207
 - Pesquisa Estendida 321, 323
 - XML 267

arquivos com estrutura de tabela

- acessando com o DB2
 - Information Integrator 152

exemplo 151

incluindo em um sistema

- federado
 - registrando o invólucro 155
 - registrando o servidor 156
 - registrando pseudônimos 157

mensagens 160

modelo de controle de acesso ao

arquivo 159

otimização 160

pseudônimos, objetos válidos

- para 13
- tipos 152
- versões suportadas 3
- visão geral 151

Arquivos do Excel

- cenário de usuário de amostra 208
- descrição 203
- incluindo em um sistema federado
 - instrução CREATE NICKNAME 207

Arquivos do Excel (continuação)

- incluindo em um sistema federado (continuação)
 - instrução CREATE SERVER 206
 - registrando o invólucro 205
 - registrando o servidor 206
 - registrando pseudônimos 207
- mensagens 211
- modelo de controle de acesso ao arquivo 210
- pseudônimos, objetos válidos para 13
- versões suportadas 3

arquivos planos

- Consulte arquivos com estrutura de tabela 3

atualizando estatísticas, planejando 22

B

BLAST

- descrição 219
- incluindo em um sistema federado
 - arquivo de configuração do BLAST 225
 - iniciando o daemon do BLAST 229
 - instalando e configurando o daemon do BLAST 225
 - instrução CREATE NICKNAME 232
 - instrução CREATE SERVER 231
 - instrução CREATE WRAPPER 230
 - registrando o invólucro 230
 - registrando o servidor 231
 - registrando pseudônimos 232
 - verificando se o executável blastall correto está instalado 225
 - verificando se os arquivos matriz corretos estão instalados 225
 - mensagens 243

BLAST (continuação)

- pseudônimos, objetos válidos para 13
- versões suportadas 3

C

campos mapeados

- Pesquisa Estendida 312
- catálogo
 - Consulte o catálogo global 387
- catálogo global

exibições contendo informações federadas 387

Centro de Comandos

- configurando origens de dados 1

Centro de Controle

- configurando origens de dados 1

configurações

- origens de dados federadas
- visão geral 8

configurando

- HMMER daemon 338

D

DB2 para iSeries

- mapeamentos de tipos de encaminhamento padrão 413
- mapeamentos de tipos reversos padrão 433
- pseudônimos, objetos válidos para 13
- tipos de servidor válidos 409
- versões suportadas 3

DB2 para Linux, UNIX e Windows

- mapeamentos de tipos de encaminhamento padrão 413
- mapeamentos de tipos reversos padrão 433
- pseudônimos, objetos válidos para 13
- tipos de servidor válidos 409
- versões suportadas 3

DB2 para VM e VSE

- mapeamentos de tipos de encaminhamento padrão 413
- mapeamentos de tipos reversos padrão 433

DB2 para VM e VSE (*continuação*)
 pseudônimos, objetos válidos para 13
 tipos de servidor válidos 409
 versões suportadas 3

DB2 para z/OS e OS/390
 mapeamentos de tipos de encaminhamento padrão 413
 mapeamentos de tipos reversos padrão 433
 pseudônimos, objetos válidos para 13
 tipos de servidor válidos 409
 versões suportadas 3

definições de servidor
 descrição 10

deteção de problemas
 configurações do Microsoft SQL Server 108
 configurações do ODBC 124
 configurações do Oracle 76
 configurações do Sybase 92
 configurações do Teradata 141

distinção entre maiúsculas e minúsculas
 preservando valores que fazem distinção entre maiúsculas e minúsculas 20

Documentum
 acesso de usuário a documentos 194
 definição dupla dos atributos de repetição 193
 descrição 165
 exemplo 165
 incluindo em um sistema federado
 apontando para o arquivo dmcl.ini do cliente Documentum 169
 disponibilizando bibliotecas do cliente do Documentum para o DB2 (apenas Ambiente Operacional AIX e Solaris) 168
 instrução CREATE FUNCTION 181
 instrução CREATE NICKNAME 174
 instrução CREATE SERVER 172
 Instrução CREATE USER MAPPING 173
 instrução CREATE WRAPPER 171

Documentum (*continuação*)
 incluindo em um sistema federado (*continuação*)
 mapeando usuários 173
 registrando funções personalizadas 181
 registrando o invólucro 171
 registrando o servidor 172
 registrando pseudônimos 174
 utilitário CreateNicknameFile 189
 mensagens 194
 pseudônimos, objetos válidos para 13
 utilitário CreateNicknameFile 189
 versões suportadas 3

E

Entrez
 pseudônimos, objetos válidos para 13
 versões suportadas 3

especificações de índice
 descrição 17

estatísticas
 atualizando origem de dados 22

exibições do catálogo SYSCAT 387
 exibições do catálogo SYSSTAT 387

F

funções personalizadas
 Pesquisa Estendida 320

G

gabaritos de funções
 Pesquisa Estendida 320

H

HMMER
 consultas
 construir 350
 invólucro 333
 pseudônimos, objetos válidos para 13
 versões suportadas 3

I

idioma de consulta generalizado
 Pesquisa Estendida 326

IDs de usuário
 valores com distinção entre maiúsculas e minúsculas 20
 incluindo
 HMMER, federado 337

Informix
 ajuste 60
 configurando acesso a 47
 configurando arquivos de configuração do cliente 48
 incluindo em um sistema federado 47
 criando mapeamentos de usuário 54
 instrução CREATE NICKNAME 59
 instrução CREATE SERVER 52
 instrução CREATE USER MAPPING 55
 registrando pseudônimos 58
 testando conexões 56

mapeamentos de tipos de encaminhamento padrão 413
 mapeamentos de tipos reversos padrão 433
 pseudônimos, objetos válidos para 13
 registrando definições de servidor 51
 registrando wrappers 50
 testando arquivos de configuração do cliente 48
 tipos de servidor válidos 409
 versões suportadas 3

início
 HMMER
 daemon 341

instrução CREATE FUNCTION
 Documentum 181
 Pesquisa Estendida 320, 357

instrução CREATE NICKNAME
 arquivos com estrutura de tabela 157
 Arquivos do Excel 207
 BLAST 232
 Documentum 174
 Exemplos 5
 Informix 58, 59
 Microsoft SQL Server 106, 107
 ODBC 122, 123
 Oracle 74, 75
 Origens de Dados da Família DB2 42, 43
 Pesquisa Estendida 317, 366
 Sybase 90, 91
 Teradata 139, 140
 XML 260, 261, 375

- instrução CREATE SERVER
 - arquivos com estrutura de tabela 156
 - Arquivos do Excel 206
 - BLAST 231
 - Documentum 172
 - Informix 52
 - Microsoft SQL Server 101
 - ODBC 117
 - OLE DB 148
 - Oracle 69
 - Origens de Dados da Família DB2 37
 - Pesquisa Estendida 316, 383
 - Sybase 84
 - Teradata 132, 134
 - XML 254
 - Instrução CREATE USER MAPPING
 - Documentum 173
 - Informix 54
 - Microsoft SQL Server 103, 104
 - ODBC 119
 - OLE DB 149
 - Oracle 71
 - Origens de Dados da Família DB2 38, 39
 - Pesquisa Estendida 318, 385
 - Sybase 87
 - Teradata 135, 136
 - instrução CREATE WRAPPER
 - arquivos com estrutura de tabela 155
 - Arquivos do Excel 205
 - BLAST 230
 - Documentum 171
 - ODBC 115
 - Pesquisa Estendida 316, 386
 - XML 252
 - instrução SET SERVER OPTION
 - definindo uma opção temporariamente 10
- M**
- mapeamentos de funções
 - descrição 16
 - Planejamento 24
 - mapeamentos de tipos de avanço
 - mapeamentos padrão 413
 - mapeamentos de tipos de dados
 - avanço 413
 - descrição 15
 - Planejamento 22
 - reverso 433
 - mapeamentos de tipos reversos
 - mapeamentos padrão 433
- O**
- objetos
 - nomeação 19
 - objetos da origem de dados
 - descrição 12
 - local 5
 - mapeamentos de usuário
 - descrição 11
 - Pesquisa Estendida 318, 385
 - Planejamento 25
 - mapeando usuários
 - Pesquisa Estendida 318, 385
 - mensagens
 - Invólucro do documento 194
 - invólucro do Excel 211
 - wrapper do arquivo estruturado de tabela 160
 - wrapper do BLAST 243
 - wrapper do XML 243
 - wrapper HMMER 352
 - wrapper Pesquisa Estendida 328
 - Microsoft Excel
 - Consulte arquivos Excel 3
 - Microsoft SQL Server
 - ajuste 108
 - configurando acesso a 95
 - detecção de problemas 108
 - incluindo em um sistema federado
 - criando mapeamentos de usuário 103
 - instrução CREATE NICKNAME 107
 - instrução CREATE SERVER 101
 - Instrução CREATE USER MAPPING 104
 - registrando
 - pseudônimos 106
 - testando conexões 105
 - visão geral 95
 - mapeamentos de tipos de encaminhamento padrão 413
 - mapeamentos de tipos reversos
 - padrão 433
 - preparando servidores
 - federados 96
 - pseudônimos, objetos válidos
 - para 13
 - registrando definições de servidor 99
 - registrando wrappers 98
 - tipos de servidor válidos 409
 - versões suportadas 3
 - objetos da origem de dados
 - (*continuação*)
 - remotas 5
 - tipos de objetos válidos 13
 - objetos locais
 - descrição 5
 - objetos remotos
 - descrição 5
 - ODBC
 - ajuste 124
 - configurando acesso a 111
 - detecção de problemas 124
 - incluindo em um sistema federado
 - criando mapeamentos de usuário 119
 - instrução CREATE NICKNAME 123
 - instrução CREATE SERVER 117
 - Instrução CREATE USER MAPPING 119
 - registrando
 - pseudônimos 122
 - testando conexões 120
 - visão geral 111
 - mapeamentos de tipos de encaminhamento padrão 413
 - preparando servidores
 - federados 113
 - pseudônimos, objetos válidos
 - para 13
 - registrando definições de servidor 116
 - registrando wrappers 114
 - tipos de servidor válidos 409
 - versões suportadas 3
 - OLE DB
 - configurando acesso a 145
 - incluindo em um sistema federado
 - criando mapeamentos de usuário 149
 - instrução CREATE SERVER 148
 - Instrução CREATE USER MAPPING 149
 - visão geral 145
 - registrando definições de servidor 147
 - registrando wrappers 146
 - tipos de servidor válidos 409
 - versões suportadas 3
 - opção de coluna NUMERIC_STRING
 - definições válidas 405

opção de coluna
 VARCHAR_NO_TRAILING_
 BLANKS
 definições válidas 405

opção de mapeamento da função
 DISABLE
 definições válidas 407

opção de mapeamento de funções
 INITIAL_INSTS
 definições válidas 407

opção de mapeamento de funções
 INITIAL_IOS
 definições válidas 407

opção de mapeamento de funções
 INSTS_PER_ARGBYTE
 definições válidas 407

opção de mapeamento de funções
 INSTS_PER_INVOC
 definições válidas 407

opção de mapeamento de funções
 IOS_PER_ARGBYTE
 definições válidas 407

opção de mapeamento de funções
 IOS_PER_INVOC
 definições válidas 407

opção de mapeamento de funções
 PERCENT_ARGBYTES
 definições válidas 407

opção de mapeamento de funções
 REMOTE_NAME
 definições válidas 407

opção de servidor
 COLLATING_SEQUENCE
 definições válidas 391

opção de servidor COMM_RATE
 definições válidas 391

opção de servidor
 CONNECTSTRING
 definições válidas 391

opção de servidor CPU_RATIO
 definições válidas 391

opção de servidor DATEFORMAT
 definições válidas 391

opção de servidor
 DB2_MAXIMAL_PUSHDOWN
 definições válidas 391

opção de servidor DBNAME
 definições válidas 391

opção de servidor FOLD_ID
 definições válidas 391
 definindo em origens de dados
 do Informix 60
 valores com distinção entre
 maiúsculas e minúsculas 20

opção de servidor FOLD_PW
 definições válidas 391
 definindo em origens de dados
 do Informix 60
 valores com distinção entre
 maiúsculas e minúsculas 20

opção de servidor IFILE
 definições válidas 391

opção de servidor IGNORE_UDT
 definições válidas 391

opção de servidor
 INFORMIX_LOCK_MODE
 definições válidas 391

opção de servidor IO_RATIO
 definições válidas 391

opção de servidor
 IUD_APP_SVPT_ENFORCE
 definições válidas 391

opção de servidor
 LOGIN_TIMEOUT
 definições válidas 391

opção de servidor NODE
 definições válidas 391

opção de servidor PACKET_SIZE
 definições válidas 391

opção de servidor PASSWORD
 definições válidas 391

opção de servidor PLAN_HINTS
 definições válidas 391

opção de servidor PUSHDOWN
 definições válidas 391

opção de servidor TIMEFORMAT
 definições válidas 391

opção de servidor TIMEOUT
 definições válidas 391

opção de servidor
 TIMESTAMPFORMAT
 definições válidas 391

opção de servidor
 VARCHAR_NO_TRAILING_
 BLANKS
 definições válidas 391

opção de usuário
 REMOTE_AUTHID
 definições válidas 403

opção de usuário
 REMOTE_DOMAIN
 definições válidas 403

opção de usuário
 REMOTE_PASSWORD
 definições válidas 403

opção de usuário
 ACCOUNTING_STRING
 definições válidas 403

opções de coluna
 definições válidas 405
 descrição 14
 Exemplos 5

opções de mapeamento de funções
 definições válidas 407

opções de mapeamento de usuário
 definições válidas 403
 descrição 11

opções do servidor
 definições válidas 391
 descrição 10
 temporárias 10

Oracle
 ajuste 76
 configurando acesso a 63
 configurando arquivos de
 configuração do cliente 64
 detecção de problemas 76
 incluindo em um sistema
 federado
 criando mapeamentos de
 usuário 71
 instrução CREATE
 NICKNAME 75
 instrução CREATE
 SERVER 69
 Instrução CREATE USER
 MAPPING 71
 registrando pseudônimos 74
 testando conexões 73
 visão geral 63
 mapeamentos de tipos de
 encaminhamento padrão 413
 mapeamentos de tipos reversos
 padrão 433
 pseudônimos, objetos válidos
 para 13
 registrando definições de
 servidor 68
 registrando wrappers 65
 testando arquivos de
 configuração do cliente 64

origens de dados
 configurando 1
 etapas de configuração
 opcionais 8
 tipos de servidor válidos 409

Origens de Dados da Família DB2
 ajuste 44
 catalogando bancos de dados
 remotos 33
 catalogando entradas de nó 32
 configurando acesso a 31

Origens de Dados da Família DB2
(*continuação*)
incluindo em um sistema
federado 31
criando mapeamentos de
usuário 38
instrução CREATE
NICKNAME 43
instrução CREATE
SERVER 37
Instrução CREATE USER
MAPPING 39
registrando pseudônimos 42
testando conexões 40
registrando definições de
servidor 36
registrando wrappers 34
origens de dados não-relacionais
mapeamentos de tipos de dados,
especificando 15
otimização
arquivos com estrutura de
tabela 160
BLAST 243
HMMER
invólucro 351

P

Pesquisa Estendida
amostra de consultas 323
campos mapeados 312
criando diversos
pseudônimos 310
descrição 307
diretrizes de consulta 321
idioma de consulta
generalizado 326
incluindo em um sistema
federado
instrução CREATE
FUNCTION 320, 357
instrução CREATE
NICKNAME 317, 366
instrução CREATE
SERVER 316, 383
Instrução CREATE USER
MAPPING 318, 385
instrução CREATE
WRAPPER 316, 386
mapeando usuários 318
registrando mapeamentos do
usuário 318
registrando o invólucro 316
registrando o servidor 316

Pesquisa Estendida (*continuação*)
incluindo em um sistema
federado (*continuação*)
registrando
pseudônimos 317
Registrando uma Função 320
resumo das etapas 315
mensagens 328
origens que podem ser
pesquisadas 307
personalizando consultas 320
pseudônimos, objetos válidos
para 13
tabelas verticais 312
versões suportadas 3
pseudônimos
alterando 355
alterando um tipo de dados 355
alterando uma opção de
pseudônimo 356
descrição 12
em pseudônimos 5
em tabelas de resumo 5
Informix, criando para 58
Microsoft SQL Server, criando
para 106
objetos válidos da origem de
dados 13
ODBC, criando para 122
opções de coluna, exemplos 5
Oracle, criando para 74
origens de dado da família DB2,
criando para 42
Sybase, criando para 90
Teradata, criando para 139
visão geral de criação 5
XML, criando para 260

R

registrar pseudônimos
HMMER
origens de dados 345
registro
HMMER
invólucro 342
servidor
assistente para Redistribuir
Dados 344
regras de nomenclatura
para objetos de bancos de dados
federados 19

S

senhas
valores com distinção entre
maiúsculas e minúsculas 20
Sybase
ajuste 92
configurando acesso a 79
configurando arquivos de
configuração do cliente 80
detecção de problemas 92
incluindo em um sistema
federado
criando mapeamentos de
usuário 87
instrução CREATE
NICKNAME 91
instrução CREATE
SERVER 84
registrando pseudônimos 90
testando conexões 89
visão geral 79
mapeamentos de tipos de
encaminhamento padrão 413
mapeamentos de tipos reversos
padrão 433
pseudônimos, objetos válidos
para 13
registrando definições de
servidor 83
registrando wrappers 82
testando arquivos de
configuração do cliente 80
tipos de servidor válidos 409
versões suportadas 3

T

tabelas de resumo
criando apelidos 5
tabelas verticais
Pesquisa Estendida 312
Teradata
ajuste 141
configurando acesso a 127
detecção de problemas 141
incluindo em um sistema
federado
criando mapeamentos de
usuário 135
instrução CREATE
NICKNAME 140
instrução CREATE
SERVER 134
instrução CREATE USER
MAPPING 136

- Teradata (*continuação*)
 - incluindo em um sistema federado (*continuação*)
 - registrando definições de servidor 132
 - registrando pseudônimos 139
 - registrando wrappers 131
 - testando conexões 129, 137
 - verificando a vinculação de tempo de execução (AIX) 130
 - visão geral 127
 - mapeamentos de tipos de encaminhamento padrão 413
 - mapeamentos de tipos reversos padrão 433
 - pseudônimos, objetos válidos para 13
 - pseudônimos em servidores federados, visão geral 138
 - registro de acesso 141
 - tipos de servidor válidos 409
 - vinculação de tempo de execução, ativando (AIX) 130, 141
- tipo de dados DATALINK não suportados 15
- tipos de dados não suportados 15
- tipos de servidor tipos federados válidos 409

U

- UDTs (User-defined Types) não suportados 15
- tipos de dados não suportados 15
- utilitário CreateNicknameFile, Documentum
 - configurando 191
 - descrição 189
 - instalando 190
 - mapeando o tipo de objeto DM_ID 192

V

- variáveis de ambiente wrapper HMMER 343
- variável de ambiente DB2_DJ_COMM 156, 171, 230, 253
 - Informix, ajustando 60
 - Microsoft SQL Server, ajustando 108

- variável de ambiente DB2_DJ_COMM (*continuação*)
 - ODBC, ajustando 124
 - Oracle, ajustando 76
 - Origens de dados da família DB2, ajustando 44
 - Sybase, ajustando 92
 - Teradata, ajustando 141

W

- wrappers
 - descrição 9
 - Planejamento 25
 - Teradata, criando para 131
 - XML, criando para 252

X

- XML
 - descrição 247
 - incluindo em um sistema federado
 - criando exibições federadas para pseudônimos não-root 266
 - definindo variáveis de ambiente DB2_DJ_COMM 253
 - executando consultas 267
 - instrução CREATE NICKNAME 260, 261, 375
 - instrução CREATE SERVER 254
 - instrução CREATE WRAPPER 252
 - registrando o invólucro 252
 - registrando o servidor 254
 - registrando pseudônimos 254, 260
 - visão geral 251
 - mensagens 268
 - pseudônimos, objetos válidos para 13
 - pseudônimos e documentos de origem 254
 - recurso do modelo de custo 257, 258
 - versões suportadas 3

Entrando em Contato com a IBM

Para localizar um escritório da IBM em seu país ou região, consulte o IBM Directory of Worldwide Contacts na Web no endereço www.ibm.com/planetwide.

Informações sobre o Produto

Informações sobre o DB2 Information Integrator estão disponíveis através de telefone ou na Web.

Na Web, visite o endereço www.ibm.com/software/data/integration. Este site contém as informações mais recentes sobre a biblioteca técnica, solicitação de manuais, downloads do cliente, newsgroups, fix packs, notícias e links para recursos da Web.

Para localizar um escritório da IBM em seu país ou região, consulte o IBM Directory of Worldwide Contacts na Web no endereço www.ibm.com/planetwide.

Comentários sobre a Documentação

Sua opinião ajuda a IBM a fornecer informações de alta qualidade. Envie seus comentários sobre este manual ou outra documentação do DB2 Information Integrator. Você pode utilizar um dos seguintes métodos para enviar os comentários:

- Envie seu comentários utilizando o formulário on-line de comentários do leitor no endereço www.ibm.com/software/data/rcf.
- Envie seus comentários por e-mail (correio eletrônico) para o endereço comments@us.ibm.com. Inclua o nome do produto, o número da versão do produto e o nome e o número da peça do manual (se aplicável). Se você estiver comentando um texto específico, inclua a localização do texto (por exemplo, um título, um número de tabela ou um número de página).



Impresso em Brazil

Spine information:



IBM DB2 Information
Integrator

Guia de Configuração da Origem de Dados

Versão 8