

IBM DB2 Information Integrator



Podręcznik systemów stowarzyszonych

Wersja 8.2

IBM DB2 Information Integrator



Podręcznik systemów stowarzyszonych

Wersja 8.2

Przed skorzystaniem z tych informacji oraz produktu, którego dotyczą, należy się zapoznać z ogólnymi informacjami zamieszczonymi w części "Uwagi" na stronie 327.

Niniejszy dokument zawiera informacje dotyczące produktów firmy IBM. Jest udostępniany na warunkach umowy licencyjnej i chroniony prawem autorskim. Informacje zawarte w tej publikacji nie zawierają żadnych gwarancji dotyczących opisywanych produktów i żadnych zapisanych w niej stwierdzeń nie należy interpretować jako takich gwarancji.

Publikacje firmy IBM można zamówić elektronicznie lub u miejscowego przedstawiciela firmy IBM:

- Aby zamówić książki poprzez stronę WWW, należy skorzystać ze strony IBM Publications Center pod adresem www.ibm.com/shop/publications/order
- Aby znaleźć najbliższego lokalnego przedstawiciela firmy IBM, należy skorzystać z informacji umieszczonych na stronie IBM Directory of Worldwide Contacts pod adresem www.ibm.com/planetwide

Wysłanie informacji do firmy IBM daje jej prawo do ich używania i dystrybucji w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich nadawcy.

© Copyright International Business Machines Corporation 1998, 2004. Wszelkie prawa zastrzeżone.

Spis treści

Informacje o tej książce ix

Kto powinien przeczytać ten podręcznik ix

Część 1. Wprowadzenie 1

Rozdział 1. Przegląd informacji o systemach stowarzyszonych 3

Systemy stowarzyszone	3
Serwer stowarzyszony	4
Co to jest źródło danych?	4
Obsługiwane źródła danych	5
Stowarzyszona baza danych	7
Katalog systemowy stowarzyszonej bazy danych	8
Kompilator SQL.	8
Optymalizator zapytań	9
Kompensacja	9
Sesje tranzytowe	10
Opakowania i moduły opakowujące	11
Domyślne nazwy opakowań	12
Definicje serwerów i opcje serwera	13
Odwzorowania użytkowników	14
Pseudonimy i obiekty źródła danych	15
Poprawne obiekty źródła danych	16
Opcje kolumny pseudonimu	17
Odwzorowania typów danych	17
Odwzorowania funkcji	18
Specyfikacje indeksu	19
Kolejności zestawiania	19
Jak kolejność zestawiania wpływa na porządek sortowania	20
Konfigurowanie lokalnej kolejności zestawiania w celu optymalizacji zapytań	20
Jak przebiega interakcja z systemem stowarzyszonym	21
Procesor wiersza komend DB2 (CLP)	21
Centrum komend DB2	22
Centrum sterowania DB2.	22
Aplikacje	23
Narzędzia z rodziny DB2.	23
Dostawcy usług WWW	23

Część 2. Administrowanie i konserwacja. 25

Rozdział 2. Modyfikowanie konfiguracji źródeł danych. 27

Modyfikowanie opakowania	27
Modyfikowanie opakowania - przykłady	28
Modyfikowanie definicji serwerów i opcji serwera	28
Modyfikowanie definicji serwera i opcji serwera - informacje szczegółowe	29
Modyfikowanie wersji źródła danych w definicji serwera	29

Modyfikowanie wszystkich definicji serwerów dla określonego typu źródła danych.	30
Korzystanie z opcji serwera w definicjach serwera	31
Modyfikowanie odwzorowania użytkowników	32
Modyfikowanie pseudonimu.	34
Modyfikowanie pseudonimów - informacje szczegółowe	35
Ograniczenia dotyczące zmiany pseudonimów	35
Modyfikowanie nazw kolumn pseudonimów	37
Modyfikowanie opcji pseudonimów	38
Modyfikowanie opcji kolumn pseudonimów	39
Usuwanie opakowania	41
Usuwanie definicji serwera	42
Usuwanie odwzorowania użytkowników	43
Usuwanie pseudonimu	44

Rozdział 3. Odwzorowania typów danych. 47

Odwzorowania typów danych w systemie stowarzyszonym	47
Odwzorowania typów danych i katalog globalny stowarzyszonej bazy danych.	48
Kiedy należy tworzyć alternatywne odwzorowania typów danych	49
Odwzorowania typów danych dla nierelacyjnych źródeł danych	50
Proste i zwrotne odwzorowania typów danych	50
Tworzenie odwzorowań typów danych.	51
Tworzenie odwzorowania dla typu źródła danych – przykład.	51
Tworzenie odwzorowania dla typu danych i wersji źródła danych – przykład	52
Tworzenie odwzorowania typów dla wszystkich obiektów źródła danych na serwerze – przykład	53
Zmiana typu lokalnego dla obiektu źródła danych	54
Zmiana typu lokalnego dla obiektu źródła danych – przykłady	55
Zmiany długich typów danych na typy varchar	57

Rozdział 4. Odwzorowanie funkcji i funkcje zdefiniowane przez użytkownika 59

Odwzorowania funkcji w systemie stowarzyszonym.	59
Kiedy należy utworzyć własne odwzorowania funkcji	59
Dlaczego odwzorowania funkcji są ważne.	60
Jak działają odwzorowania funkcji w systemie stowarzyszonym	60
Wymagania dotyczące odwzorowań funkcji zdefiniowanych przez użytkownika.	61
Szablony funkcji	62
Tworzenie szablonów funkcji	62
Przekazywanie informacji o narzucie w odwzorowaniu funkcji na użytek optymalizatora zapytań	64
Modyfikowanie informacji o narzucie	65
Określanie nazw funkcji w odwzorowaniu funkcji	66
Jak utworzyć odwzorowanie funkcji	66
Definiowanie szczegółów odwzorowań funkcji	67

Tworzenie odwzorowania funkcji dla określonego typu źródła danych	67	Dostęp do nowych obiektów źródeł danych	105
Tworzenie odwzorowania funkcji dla określonego typu i wersji źródła danych.	68	Dostęp do źródeł danych przy użyciu sesji tranzytowych	107
Tworzenie odwzorowania funkcji dla wszystkich obiektów źródła danych na określonym serwerze.	69	Uzyskiwanie dostępu do danych heterogenicznych za pośrednictwem widoków stowarzyszonych	108
Funkcje zdefiniowane przez użytkownika w aplikacjach	70	Tworzenie pseudonimu dla pseudonimu	110
Wyłączanie domyślnego odwzorowania funkcji	70	Selekcja danych w systemie stowarzyszonym	110
Usuwanie odwzorowania funkcji zdefiniowanej przez użytkownika	71	Modyfikowanie danych w systemie stowarzyszonym	113
		Wstawianie danych do obiektów źródeł danych.	113
		Aktualizowanie danych w obiektach źródła danych	114
		Usuwanie danych z obiektów źródeł danych.	115
Rozdział 5. Specyfikacje indeksu	73	Semantyki przypisywania danych w systemie stowarzyszonym	116
Specyfikacje indeksów w systemie stowarzyszonym	73	Semantyki przypisań danych w systemie stowarzyszonym - przykłady	118
Tworzenie specyfikacji indeksów dla obiektów źródła danych	74		
Tworzenie specyfikacji indeksów dla tabel, do których dodano nowe indeksy	75	Rozdział 9. Monitorowanie systemu stowarzyszonego.	119
Tworzenie specyfikacji indeksów dla widoków	77	Indykatory poprawności dla stowarzyszonych pseudonimów i serwerów	119
Tworzenie specyfikacji indeksów dla synonimów Informix	78	Aktywowanie stowarzyszonych indykatorów poprawności	120
		Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów	121
		Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów - przykład	122
		Monitorowanie obrazów stanu systemów stowarzyszonych - przegląd	123
		Monitorowanie fragmentów stowarzyszonych zapytań	123
		Monitorowanie obrazów stanu fragmentów zapytań stowarzyszonych - przykład	124
Rozdział 6. Przezroczysty kod DDL.	81		
Co to jest przezroczysty kod DDL?	81	Rozdział 10. Obsługa Unicode dla stowarzyszonych źródeł danych	125
Ograniczenia w stosowaniu przezroczystego kodu DDL	82	Obsługa Unicode w systemach stowarzyszonych	125
Zdalne kolumny LOB i przezroczysty kod DDL	83	Określanie strony kodowej klienta w celu obsługi kodowania Unicode w źródłach danych Microsoft SQL Server i ODBC	127
Tworzenie nowych tabel zdalnych za pomocą przezroczystego kodu DDL	83	Strony kodowe obsługiwane przez bazę danych Unicode dla opcji CODEPAGE opakowań MSSQL i ODBC	128
Modyfikowanie tabel zdalnych za pomocą przezroczystego kodu DDL	86	Określanie strony kodowej pliku na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli	128
Usuwanie tabel zdalnych za pomocą przezroczystego kodu DDL	88	Określanie strony kodowej na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli - przykład	129
		Błędy spowodowane różnicą wielkości zdalnego i stowarzyszonego punktu kodowego	129
Rozdział 7. Obsługa transakcji w systemach stowarzyszonych	91		
Obsługa transakcji w systemie stowarzyszonym	91	Część 3. Wydajność	131
Aktualizacja w systemie stowarzyszonym.	92		
Transakcja aktualizacji w sesji tranzytowej	93	Rozdział 11. Strojenie wydajności w systemie stowarzyszonym	133
Obsługa transakcji realizowanych przy użyciu przezroczystego kodu DDL	93	Publikacje na temat wydajności systemów stowarzyszonych	133
Źródła danych automatycznie zatwierdzające instrukcje DDL	94	Strojenie przetwarzania zapytań	133
Funkcje zdefiniowane przez użytkownika przekazywane do przetwarzania w źródle danych	94	Analiza przekazywania do źródła	135
		Analiza przekazywania do źródła - informacje szczegółowe	136
Rozdział 8. Operacje INSERT, UPDATE i DELETE	95		
Uprawnienia wymagane do użycia instrukcji INSERT, UPDATE i DELETE	95		
Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym.	96		
Nieobsługiwane źródła danych	96		
Spójność referencyjna w systemie stowarzyszonym	97		
Instrukcje INSERT, UPDATE i DELETE a obiekty LOB	97		
Zachowywanie niepodzielności instrukcji w systemie stowarzyszonym	98		
Praca z pseudonimami	100		
Składnia WITH HOLD	100		
Wyzwalacze	100		
Praca z pseudonimami - informacje szczegółowe	101		
Instrukcje SQL używane w odniesieniu do pseudonimów.	101		

Cechy serwera wpływające na możliwości przekazywania	136
Cechy pseudonimów wpływające na możliwości przekazywania	140
Cechy zapytań wpływające na możliwości przekazywania	142
Decyzje wynikające z analizy przekazywania do źródła	142
Gdzie będzie wartościowane zapytanie?	142
Przebieg oceny planu dostępu	144
Uaktualnianie i dostosowywanie źródła danych	146
Globalna optymalizacja	146
Globalna optymalizacja - informacje szczegółowe	147
Cechy serwera wpływające na globalną optymalizację	147
Cechy pseudonimu wpływające na globalną optymalizację	149
Decyzje dotyczące globalnej optymalizacji	151
Globalna optymalizacja - analiza	152
Decyzje zapadające podczas optymalizacji planu dostępu	152
Elementy monitora systemu mające wpływ na wydajność	154

Rozdział 12. Paralelizm i zapytania odwołujące się do pseudonimów 157

Paralelizm i zapytania odwołujące się do pseudonimów	157
Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów	158
Włączanie paralelizmu wewnątrz partycji dla zapytań odwołujących się do pseudonimów	158
Paralelizm między partycjami i zapytania odwołujące się do pseudonimów	159
Włączanie paralelizmu między partycjami dla zapytań odwołujących się do pseudonimów	161
Obliczeniowe grupy partycji	162
Definiowanie obliczeniowej grupy partycji	162
Paralelizm między partycjami dla zapytań odwołujących się do pseudonimów - oczekiwania w zakresie wydajności	163
Paralelizm mieszany i zapytania odwołujące się do pseudonimów	164
Włączanie paralelizmu mieszanego dla zapytań odwołujących się do pseudonimów	164
Plany dostępu równoległego dla zapytań odwołujących się do pseudonimów	165
Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów - przykłady planów dostępu	165
Paralelizm między partycjami i zapytania odwołujące się do pseudonimów - przykłady planów dostępu	167
Paralelizm mieszany i zapytania odwołujące się do pseudonimów - przykłady planów dostępu	169

Rozdział 13. Zmaterializowane tabele zapytań i systemy stowarzyszone. 173

Zmaterializowane tabele zapytań i systemy stowarzyszone – przegląd	173
Tworzenie stowarzyszonej zmaterializowanej tabeli zapytania	174
Ograniczenia poszczególnych źródeł danych dotyczące zmaterializowanych tabel zapytań	175
Ograniczenia dotyczące korzystania ze zmaterializowanych tabel zapytań z pseudonimami	177

Rozdział 14. Tabele buforowane w systemie stowarzyszonym 179

Tabele buforowane	179
Tworzenie tabeli buforowanej	181
Włączanie pamięci podręcznej	182
Dodawanie zmaterializowanej tabeli zapytania do tabeli buforowanej	183
Usuwanie zmaterializowanej tabeli zapytania z tabeli buforowanej	183
Usuwanie tabeli buforowanej	184

Rozdział 15. Ograniczenia informacyjne na pseudonimach w systemie stowarzyszonym 185

Ograniczenia informacyjne pseudonimów	185
Określanie ograniczeń informacyjnych dla pseudonimów	185
Określanie ograniczeń informacyjnych dla pseudonimów - przykłady	186

Rozdział 16. Statystyka pseudonimu 191

Narzędzie do aktualizacji statystyk pseudonimu - przegląd	191
Pobieranie statystyk pseudonimów	192
Pobieranie statystyk pseudonimów z wiersza komend - przykłady	194
Tworzenie katalogu narzędzi DB2.	194
Wyświetlanie statusu aktualizacji statystyk pseudonimu	195

Część 4. Programowanie aplikacji 197

Rozdział 17. Scenariusz programowania aplikacji. 199

Rozdział 18. Programowanie aplikacji dla systemów stowarzyszonych 201

Jak aplikacje klienckie współpracują ze źródłami danych	201
Praca z pseudonimami w aplikacjach	202
Odwoływanie się do obiektów źródeł danych za pośrednictwem pseudonimów w instrukcjach SQL	202
Pseudonimy w instrukcjach DDL	202
Wpływ statystyk źródła danych na aplikacje.	203
Pseudonimy wywołujące procedury zapisane w bazie	205
Definiowanie opcji kolumn dla pseudonimów	205
Tworzenie i używanie widoków stowarzyszonych	206
Korzystanie z poziomów odseparowania do podtrzymania spójności danych	207
Stowarzyszona obsługa obiektów LOB	208
Stowarzyszona obsługa obiektów LOB - informacje szczegółowe	210
Wskaźniki LOB	210
Ograniczenia dotyczące obiektów LOB	210
Rozproszone żądania kierujące zapytania do źródeł danych.	210
Optymalizacja żądań rozproszonych przy użyciu opcji serwera	212
Korzystanie z sesji tranzytowych w aplikacjach.	213
Bezpośrednie kierowanie zapytań do źródeł danych w sesji tranzytovej	213

Uwagi i ograniczenia dotyczące sesji tranzytowych w systemach stowarzyszonych	214
Sesje tranzytowe do źródeł danych Oracle	214

Część 5. Informacje dodatkowe 217

Rozdział 19. Widoki w tabeli katalogu globalnego zawierające informacje o systemie stowarzyszonym	219
--	------------

Rozdział 20. Opcje opakowań w systemie stowarzyszonym	223
--	------------

Rozdział 21. Opcje serwera w systemie stowarzyszonym	225
---	------------

Rozdział 22. Opcje odwzorowań użytkowników w systemach stowarzyszonych	241
---	------------

Rozdział 23. Opcje pseudonimu dla systemów stowarzyszonych	243
---	------------

Rozdział 24. Opcje kolumn pseudonimu dla systemów stowarzyszonych	253
--	------------

Rozdział 25. Opcje odwzorowań funkcji w systemach stowarzyszonych	261
--	------------

Rozdział 26. Poprawne typy serwerów w instrukcjach SQL	263
---	------------

Opakowanie BioRS	263
Opakowanie BLAST	263
Opakowanie CTLIB	264
Opakowanie Documentum	264
Opakowanie DRDA	264
Opakowanie Entrez	265
Opakowanie Excel	265
Opakowanie Extended Search	265
Opakowanie HMMER	265
Opakowanie Informix	265
Opakowanie MSSQLODBC3	266
Opakowanie NET8	266
Opakowanie ODBC	266
Opakowanie OLE DB	266
Opakowanie pliku o strukturze tabeli	266
Opakowanie Teradata	266
Opakowanie usług WWW	267
Opakowanie WebSphere Business Integration	267
Opakowanie XML	267

Rozdział 27. Domyślne proste odwzorowania typów	269
--	------------

Źródła danych DB2 for z/OS and OS/390	269
Źródła danych DB2 dla iSeries.	270

Źródła danych DB2 Server dla VM i VSE	271
Źródła danych DB2 dla systemów Linux, UNIX i Windows	272
Źródła danych Informix	274
Źródła danych Microsoft SQL Server.	275
Źródła danych ODBC	278
Źródła danych Oracle NET8	279
Źródła danych Sybase	280
Źródła danych Teradata	281

Rozdział 28. Domyślne zwrotne odwzorowania typów	285
---	------------

Źródła danych DB2 for z/OS and OS/390	286
Źródła danych DB2 dla iSeries.	287
Źródła danych DB2 dla VM i VSE	288
Źródła danych DB2 dla systemów Linux, UNIX i Windows	289
Źródła danych Informix	290
Źródła danych Microsoft SQL Server.	291
Źródła danych Oracle NET8	292
Źródła danych Sybase	293
Źródła danych Teradata	294

Rozdział 29. Domyślne odwzorowania typów danych dla bazy danych Unicode.	297
---	------------

Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie NET8	297
Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie NET8	297
Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie Sybase	298
Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie Sybase	298
Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie ODBC	299
Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie ODBC	299
Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie Microsoft SQL Server	300
Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie Microsoft SQL Server	300

Rozdział 30. Typy danych obsługiwane dla nierelacyjnych źródeł danych	301
--	------------

Typy danych obsługiwane przez opakowanie BioRS	301
Typy danych obsługiwane przez opakowanie BLAST	301
Typy danych obsługiwane przez opakowanie Documentum	302
Typy danych obsługiwane przez opakowanie Entrez	302
Typy danych obsługiwane przez opakowanie Excel	303
Typy danych obsługiwane przez opakowanie Extended Search	303
Typy danych obsługiwane przez opakowanie HMMER	303
Typy danych obsługiwane przez opakowanie dla plików o strukturze tabeli	304
Typy danych obsługiwane przez opakowanie dla usług WWW	304

	Typy danych obsługiwane przez opakowanie WebSphere Business Integration	305			
	Typy danych obsługiwane przez opakowanie XML	305			
	Rozdział 31. Elementy monitora systemu stowarzyszonej bazy danych	307			
	Rozdział 32. Procedura zapisana w bazie SYSPROC.NNSTAT	309			
	Rozdział 33. Usuwanie skutków awarii w środowiskach o wysokiej dostępności a stowarzyszone źródła danych	311			
	Rozdział 34. Informacje o serwerze bramy zapytań dla pułapek mechanizmu	313			
	Dokumentacja techniczna programu DB2 Information Integrator.	315			
	Dokumentacja programu DB2 Information Integrator	315			
	Dostęp do dokumentacji programu DB2 Information Integrator	315			
	Dokumentacja funkcji replikacji w systemie z/OS	317			
	Dokumentacja funkcji publikowania zdarzeń programu DB2 Universal Database w systemie z/OS	318			
	Dokumentacja funkcji publikowania zdarzeń dla IMS i VSAM w systemie z/OS	319			
	Dokumentacja funkcji publikowania zdarzeń i replikacji w systemach Linux, UNIX i Windows	319			
					Dokumentacja funkcji stowarzyszonej w systemie z/OS
					320
					Dokumentacja funkcji stowarzyszonej w systemach Linux, UNIX i Windows
					321
					Dokumentacja funkcji Enterprise Search w systemach Linux, UNIX i Windows
					322
					Uwagi do wydania oraz wymagania instalacyjne
					322
					Uwagi do wydania oraz wymagania instalacyjne
					323
					Pakiety poprawek do dokumentacji programu DB2 Information Integrator
					324
					Ułatwienia dostępu
					325
					Wprowadzanie danych i nawigacja za pomocą klawiatury
					325
					Operowanie programem za pomocą klawiatury
					325
					Nawigacja przy użyciu klawiatury.
					325
					Miejsce aktywne dla klawiatury
					325
					Przystępny ekran
					325
					Ustawienia czcionek
					326
					Niezależność od kolorów
					326
					Zgodność z rozwiązaniami technicznymi dla niepełnosprawnych
					326
					Dokumentacja w przystępnym formacie
					326
					Uwagi.
					327
					Znaki towarowe
					329
					Indeks
					331
					Kontakt z firmą IBM.
					337
					Informacje o produkcie
					337
					Komentarze do dokumentacji
					337

Informacje o tej książce

W tym podręczniku opisano zasady pracy w systemie stowarzyszonym po zainstalowaniu serwera stowarzyszonego i skonfigurowaniu dostępu do źródeł danych.

W podręczniku przedstawiono następujące tematy:

- Wprowadzenie do pojęć dotyczących systemów stowarzyszonych, komponentów i możliwości tych systemów.
- Instrukcje dotyczące modyfikowania instalacji serwera stowarzyszonego i konfiguracji źródeł danych.
- Objasnienie obsługi transakcji w systemach stowarzyszonych.
- Zalecenia dotyczące administrowania serwerem stowarzyszonym i strojenia go w celu uzyskania optymalnej wydajności.
- Objasnienie zagadnień, które należy wziąć pod uwagę podczas tworzenia aplikacji dla systemu stowarzyszonego.
- Obszerne informacje uzupełniające o każdym źródle danych.

Pionowe linie na lewym marginesie stron oznaczają zmiany techniczne wprowadzone w tekście tego podręcznika.

Kto powinien przeczytać ten podręcznik

Podręcznik ten jest przeznaczony dla administratorów systemu, administratorów baz danych, administratorów ochrony i operatorów systemu, którzy zajmują się instalowaniem, konfigurowaniem i konserwacją systemów stowarzyszonych albo którzy korzystają z systemów stowarzyszonych. Lektura tego podręcznika pozwoli uzyskać w systemie stowarzyszonym dostęp do danych z relacyjnych i nierelacyjnych źródeł danych. Z tego podręcznika korzystać mogą także programiści i inni użytkownicy, którzy chcą poznać zasady konfigurowania systemu stowarzyszonego, administrowania takim systemem i korzystania z niego.

Zakłada się, że czytelnik tej książki zna program DB2 UDB. Wymagana jest znajomość standardowej terminologii dotyczącej baz danych i doświadczenie w projektowaniu baz danych i administrowaniu nimi. Zakłada się ponadto znajomość używanych aplikacji i źródeł danych, które mają być używane przez systemy stowarzyszone.

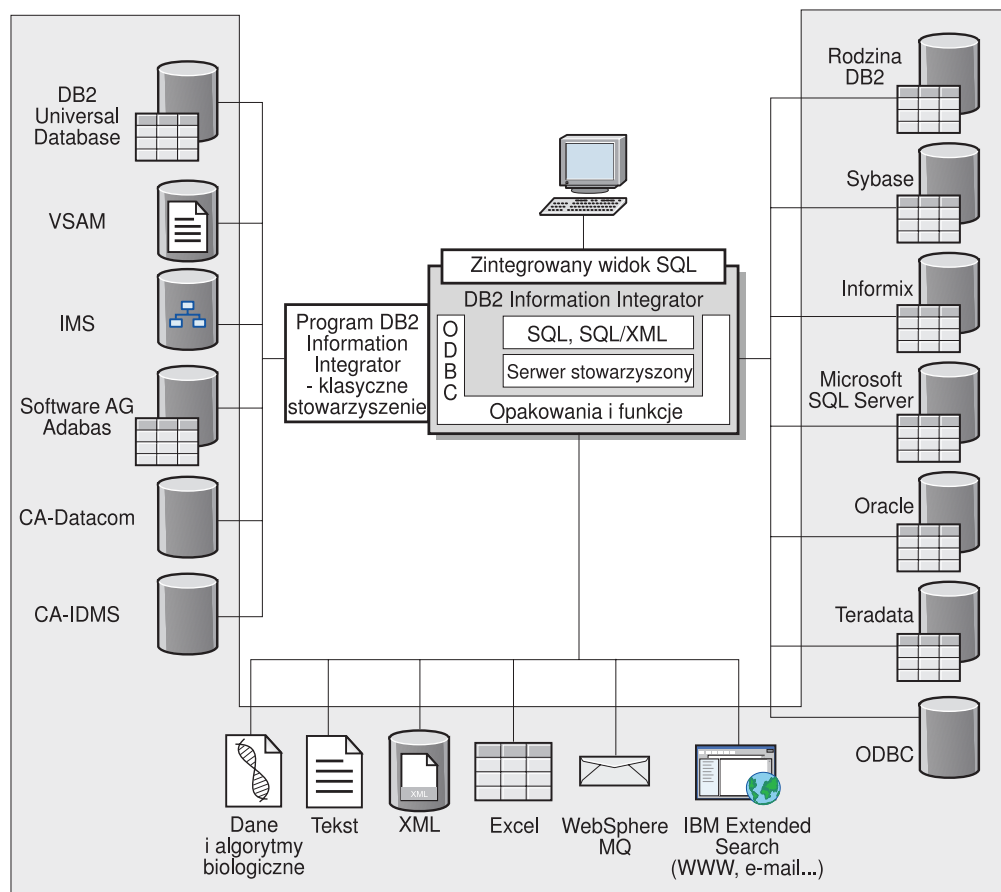
Część 1. Wprowadzenie

Rozdział 1. Przegląd informacji o systemach stowarzyszonych

W tym rozdziale opisano opcje systemu stowarzyszonego, zdefiniowano koncepcje i terminologię używaną w tym podręczniku oraz naszkicowano metody pracy z systemami stowarzyszonymi.

Systemy stowarzyszone

System stowarzyszony DB2[®] jest specjalnym typem rozproszonego systemu zarządzania bazami danych (DBMS). Na system stowarzyszony składają się: instancja DB2 działająca na serwerze stowarzyszonym, baza danych pełniąca rolę stowarzyszonej bazy danych, przynajmniej jedno źródło danych oraz klienci (użytkownicy i aplikacje) korzystający z bazy danych i ze źródeł danych. W systemie stowarzyszonym istnieje możliwość wysyłania żądań rozproszonych do wielu źródeł danych za pomocą jednej instrukcji SQL. Przykładem może być łączenie ze sobą danych znajdujących się w tabeli DB2 Universal Database[™], w tabeli Oracle i w pliku ze znacznikami XML w ramach jednej instrukcji SQL. Poniższy rysunek przedstawia komponenty systemu stowarzyszonego oraz przykłady źródeł danych, z których można korzystać.



Rysunek 1. Komponenty systemu stowarzyszonego

Potencjał systemów stowarzyszonych DB2 wyraża się następującymi możliwościami:

- Łączenie danych z tabel lokalnych i zdalnych źródeł danych w taki sposób, jak gdyby wszystkie te dane były przechowywane w lokalnej stowarzyszonej bazie danych.
- Aktualizowanie danych w relacyjnych źródłach danych, tak jak gdyby dane te znajdowały się w stowarzyszonej bazie danych.
- Replikowanie danych do i z relacyjnych źródeł danych.
- Możliwość wykorzystania mocy obliczeniowej źródeł danych przez zlecenie wykonania w źródle danych fragmentów żądań.
- Kompensowanie ograniczeń implementacji języka SQL w źródłach danych przez przetwarzanie fragmentów żądań rozproszonych na serwerze stowarzyszonym.

Serwer stowarzyszony

Serwer DB2[®] w systemie stowarzyszonym jest określany mianem *serwera stowarzyszonego*. Do pracy w charakterze serwerów stowarzyszonych można skonfigurować dowolną liczbę instancji DB2. Mogą to być istniejące instancje DB2 lub nowe instancje, utworzone specjalnie w tym celu.

Instancja DB2 zarządzająca systemem stowarzyszonym jest nazywana *serwerem*, ponieważ odpowiada ona na żądania kierowane przez użytkowników i aplikacje klienckie. Serwer stowarzyszony często przesyła fragmenty otrzymanych zapytań do źródeł danych, zlecając im ich przetwarzanie. Operacja *przekazana do źródła* jest to operacja wykonywana na zdalnym serwerze. Instancja DB2 zarządzająca systemem stowarzyszonym jest nazywana *serwerem stowarzyszonym* mimo że delegując żądania do źródeł danych sama pełni rolę klienta.

Jak każdy inny serwer aplikacji, serwer stowarzyszony jest instancją menedżera bazy danych. Procesy aplikacji nawiązują połączenia z bazą danych i wysyłają do niej żądania za pośrednictwem serwera stowarzyszonego. Są jednak dwie opcje odróżniające go od innych serwerów aplikacji:

- Serwer stowarzyszony może odbierać żądania, które częściowo lub w całości muszą być realizowane przez źródła danych. Serwer stowarzyszony przesyła takie żądania do źródeł danych.
- Podobnie jak inne serwery aplikacji, serwer stowarzyszony korzysta z protokołów DRDA[®] (na warstwie TCP/IP) do komunikowania się z instancjami rodziny DB2. Jednak w przeciwieństwie do nich serwer stowarzyszony odwołuje się do źródła danych za pośrednictwem rodzimego klienta odpowiedniego źródła danych. Na przykład dostęp do źródeł Sybase odbywa się za pomocą klienta Sybase Open Client, a dostęp do źródeł Microsoft[®] SQL Server realizowany jest poprzez sterownik Microsoft SQL Server ODBC.

Pojęcia pokrewne:

- “Co to jest źródło danych?” na stronie 4

Co to jest źródło danych?

W systemie stowarzyszonym *źródłem danych* może być instancja relacyjnej bazy danych (na przykład Oracle lub Sybase) lub źródło nierelacyjne (na przykład algorytm wyszukiwania BLAST lub plik ze znacznikami XML). Niektóre źródła danych dają dostęp do innych źródeł danych. Na przykład źródło danych Extended Search umożliwia korzystanie z takich źródeł danych, jak bazy danych Lotus[®] Notes, Microsoft[®] Access, Microsoft Index Server, wyszukiwarki internetowe oraz katalogi LDAP (Lightweight Directory Access Protocol).

Metoda lub protokół używane podczas dostępu do źródła danych zależą od typu źródła danych. Na przykład protokół DRDA[®] jest stosowany przy dostępie do źródeł danych DB2[®] for z/OS[™] and OS/390[®], a biblioteka/API klienta Documentum umożliwia dostęp do źródeł danych Documentum.

Źródła danych działają na wół autonomicznie. Na przykład serwer stowarzyszony może wysłać zapytania do źródła danych Oracle i korzystać zeń równocześnie z pracującymi niezależnie od niego dedykowanymi aplikacjami Oracle. Serwer stowarzyszony DB2 nie monopolizuje ani nie ogranicza dostępu do innych źródeł danych poza przypadkami stosowania ograniczeń spójności i blokad.

Pojęcia pokrewne:

- “Stowarzyszona baza danych” na stronie 7

Informacje pokrewne:

- “Obsługiwane źródła danych” na stronie 5

Obsługiwane źródła danych

System stowarzyszony umożliwia dostęp do wielu rodzajów źródeł danych. W poniższej tabeli zamieszczono wykaz obsługiwanych źródeł:

Tabela 1. Obsługiwane wersje źródeł danych i metody dostępu.

Źródło danych	Obsługiwane wersje	Metoda dostępu
DB2 Universal Database [™] dla systemów Linux, UNIX i Windows [®]	7.2, 8.1, 8.2	DRDA [®]
DB2 Universal Database for z/OS [™] and OS/390 [®]	6.1, 7.1 z następującymi poprawkami APAR: <ul style="list-style-type: none"> • PQ62695 • PQ55393 • PQ56616 • PQ54605 • PQ46183 • PQ62139 8.1	DRDA
DB2 Universal Database for iSeries [™]	5.1 <ul style="list-style-type: none"> • z następującymi poprawkami APAR: <ul style="list-style-type: none"> – SE06003 – SE06872 – II13348 • z następującymi poprawkami PTF: <ul style="list-style-type: none"> – SI05990 SI05991 5.2 z zastosowaną poprawką PTF SI0735.	DRDA

Tabela 1. Obsługiwane wersje źródeł danych i metody dostępu. (kontynuacja)

Źródło danych	Obsługiwane wersje	Metoda dostępu
DB2 Server for VM and VSE	7.1 (lub nowsza) z poprawkami APAR dla funkcji schematu.	DRDA
Informix™	7.31, 8.32, 8.4, 9.3, 9.4	Informix Client SDK, wersja 2.7 (lub nowsza)
ODBC	3.x	Sterownik ODBC dla źródła danych, na przykład sterownik Redbrick ODBC udostępniający źródła Redbrick.
OLE DB	2.7, 2.8	OLE DB 2.0 (lub nowsza)
Oracle	8.0.6, 8.1.6, 8.1.7, 9.0, 9.1, 9.2, 9i, 10g	Oprogramowanie klienta Oracle net lub NET8
Microsoft SQL Server	Wersja 7.0, z dodatkiem 2000 SP3 i nowszymi dla tej wersji	W systemie Windows, sterownik Microsoft SQL Server Client ODBC 3.0 (lub nowszy). W systemie UNIX, sterownik DataDirect Technologies (dawniej MERANT) Connect ODBC 3.7 (lub nowszy).
Sybase	11.9.2, 12.x	Interfejs Sybase Open Client ctlib
Teradata	V2R3, V2R4, V2R5	Teradata Call-Level Interface, wersja 2 (CLIV2), wydanie 04.06 (lub nowsze)
BLAST	Obsługiwane pakiety poprawek 2.2.3 i nowsze 2.2	Demon BLAST (dostarczany wraz z opakowaniem)
BioRS	Wersja 5.0.14	Brak
Documentum	3.x, 4.x	Documentum Client library/APL, wersja 3.1.7a (lub nowsza)
Entrez (źródła danych PubMed i GenBank)	1.0	Brak
HMMER	2.2g, 2.3	Demon HMMER (dostarczany wraz z opakowaniem)
IBM Lotus Extended Search	4.0.1, 4.0.2	Biblioteka klienta Extended Search (dostarczana wraz z opakowaniem)
Microsoft Excel	97, 2000, 2002, 2003	Programy Excel 97, 2000, 2002 lub 2003 zainstalowane na serwerze stowarzyszonym
PeopleSoft	8.x	IBM WebSphere Business Integration Adapter for PeopleSoft, wersje 2.3.1 i 2.4
SAP	3.x, 4.x	IBM WebSphere Business Integration Adapter for mySAP.com, wersje 2.3.1 i 2.4

Tabela 1. Obsługiwane wersje źródeł danych i metody dostępu. (kontynuacja)

Źródło danych	Obsługiwane wersje	Metoda dostępu
Siebel	7, 7.5, 2000	IBM WebSphere Business Integration Adapter for Siebel eBusiness Applications, wersje 2.3.1 i 2.4
Pliki o strukturze tabeli		Brak
Funkcje zdefiniowane przez użytkownika dla KEGG	Obsługiwane	
Funkcje zdefiniowane przez użytkownika dla Life Sciences	Obsługiwane	
Usługi WWW	Specyfikacje SOAP 1.0., 1.1, WSDL 1.0, 1.1	HTTP
XML	Specyfikacja 1.0	Brak

Pojęcia pokrewne:

- “Co to jest źródło danych?” na stronie 4

Stowarzyszona baza danych

Od strony użytkowników i aplikacji klienckich źródła danych widoczne są jako pojedyncza, zbiorowa baza danych DB2[®]. Użytkownicy i aplikacje komunikują się ze *stowarzyszoną bazą danych*, którą zarządza serwer stowarzyszony. W stowarzyszonej bazie danych zawarty jest katalog systemowy. Katalog systemu stowarzyszonej bazy danych zawiera wpisy identyfikujące źródła danych i opisujące ich cechy. Serwer stowarzyszony korzysta z informacji zapisanych w katalogu systemu stowarzyszonej bazy danych oraz w opakowaniach źródeł danych w celu ustalenia najlepszego planu przetwarzania instrukcji SQL.

System stowarzyszony przetwarza instrukcje SQL tak, jak gdyby źródła danych były zwykłymi tabelami lub widokami relacyjnymi w stowarzyszonej bazie danych. W efekcie:

- System stowarzyszony umożliwia łączenie danych relacyjnych z danymi w formatach nierelacyjnych. Jest to możliwe nawet w przypadku źródeł danych obsługujących inne dialekty SQL lub całkiem pozbawionych obsługi SQL.
- Właściwości stowarzyszonej bazy danych mają wyższy priorytet w przypadku rozbieżności między charakterystyką stowarzyszonej bazy danych a charakterystyką źródła danych:
 - Założmy, że strona kodowa serwera stowarzyszonego jest inna niż strona kodowa używana przez źródło danych. Dane znakowe zwracane użytkownikowi ze źródła danych są przekształcane zgodnie ze stroną kodową używaną przez stowarzyszoną bazę danych.
 - Założmy, że kolejność zestawiania serwera stowarzyszonego jest inna niż kolejność zestawiania używana przez źródło danych. Wszelkie operacje sortowania są wykonywane na serwerze stowarzyszonym, a nie w źródle danych.

Pojęcia pokrewne:

- “Kompilator SQL” na stronie 8
- “Katalog systemowy stowarzyszonej bazy danych” na stronie 8

Katalog systemowy stowarzyszonej bazy danych

Katalog systemowy stowarzyszonej bazy danych zawiera informacje dotyczące obiektów w stowarzyszonej bazie danych oraz obiektów w źródłach danych. Katalog w stowarzyszonej bazie danych jest nazywany *katalogiem globalnym*, ponieważ zawiera on informacje o całym systemie stowarzyszonym. Optymalizator zapytań DB2[®] korzysta z informacji zapisanych w katalogu globalnym i w opakowaniu źródła danych w celu zaplanowania najlepszej metody przetwarzania instrukcji SQL. Katalog globalny zawiera informacje dotyczące lokalnych i zdalnych nazw kolumn, typów danych kolumn, wartości domyślnych kolumn oraz indeksów.

Informacje *zdalne* w katalogu są to nazwy i informacje używane przez źródła danych. Informacje *lokalne* w katalogu są to nazwy i informacje używane przez stowarzyszoną bazę danych. Dla przykładu założmy, że tabela zdalna zawiera kolumnę o nazwie *EMPNO*. W katalogu globalnym zdalna kolumna zapisana byłaby również pod nazwą *EMPNO*. Także lokalna nazwa kolumny zostanie zapisana jako *EMPNO*, chyba że użytkownik poda inną nazwę. Jednak lokalną nazwę kolumny można zmienić na przykład na *Numer_Pracownika*. Użytkownicy tworzący zapytania względem tej kolumny będą mogli korzystać z nazwy *Numer_Pracownika* zamiast nazwy *EMPNO*. Do zmiany lokalnych nazw kolumn źródła danych służy instrukcja ALTER NICKNAME.

W przypadku relacyjnych źródeł danych informacje w katalogu globalnym opisują zarówno parametry lokalne, jak i zdalne.

W przypadku źródeł nierelacyjnych zawartość katalogu globalnego zależy od źródła danych.

Aby wyświetlić zapisane w katalogu globalnym informacje na temat zdalnych tabel, należy utworzyć zapytania względem widoków katalogowych SYSCAT.TABLES, SYSCAT.TABOPTIONS, SYSCAT.INDEXES, SYSCAT.COLUMNS i SYSCAT.COLOPTIONS w stowarzyszonej bazie danych.

Katalog globalny zawiera także inne informacje na temat źródła danych. Na przykład mogą to być informacje umożliwiające nawiązanie połączenia między serwerem stowarzyszonym a źródłem danych oraz odwzorowanie uprawnień użytkowników serwera stowarzyszonego na uprawnienia użytkowników w źródle danych. Katalog globalny zawiera wartości atrybutów źródła danych ustawione jawnie, takie jak opcje serwera.

Pojęcia pokrewne:

- “Kompilator SQL” na stronie 8

Informacje pokrewne:

- Rozdział 19, “Widoki w tabeli katalogu globalnego zawierające informacje o systemie stowarzyszonym”, na stronie 219

Kompilator SQL

W celu pobrania danych ze źródeł danych użytkownicy i aplikacje wprowadzają zapytania w języku DB2[®] SQL do stowarzyszonej bazy danych. Po odebraniu zapytania kompilator DB2 SQL analizuje informacje zawarte w katalogu globalnym oraz w opakowaniu źródła danych, aby usprawnić realizację zapytania. Są to informacje dotyczące łączenia się ze źródłem danych, atrybutów serwera, odwzorowań, indeksu i statystyk przetwarzania.

Pojęcia pokrewne:

- “Opakowania i moduły opakujące” na stronie 11
- “Optymalizator zapytań” na stronie 9

Optymalizator zapytań

W procesie kompilacji SQL zapytanie jest analizowane przez *optymalizator zapytań*. Kompilator opracowuje alternatywne strategie przetwarzania zapytania zwane *planami dostępu*. Plan dostępu może przewidywać, że zapytanie będzie:

- przetwarzane przez źródło danych,
- przetwarzane przez serwer stowarzyszony,
- przetwarzane częściowo przez źródło danych i częściowo przez serwer stowarzyszony.

Program DB2[®] UDB ocenia plan dostępu przede wszystkim na podstawie informacji o możliwościach źródła danych i o samych danych. Odpowiednie informacje zawarte są w opakowaniu oraz w katalogu globalnym. Program DB2 UDB rozkłada zapytanie na segmenty nazywane *fragmentami zapytania*. Zazwyczaj efektywniejsze jest *przekazanie* fragmentu zapytania do źródła danych, o ile źródło danych może go przetworzyć. Jednak optymalizator zapytań uwzględnia jeszcze inne czynniki, takie jak:

- ilość danych, jaka musi być przetworzona,
- szybkość systemu źródła danych,
- ilość danych zwracanych w wyniku przetworzenia danego fragmentu,
- szerokość pasma komunikacyjnego,
- czy na serwerze stowarzyszonym istnieje zmaterializowana tabela zapytania reprezentująca wynik tego samego zapytania.

Optymalizator zapytań generuje plany dostępu lokalnego i zdalnego dla fragmentu zapytania na podstawie kosztu zasobów. Następnie program DB2 UDB wybiera plan dostępu, dla którego koszt wykonania zapytania jest najniższy.

Jeśli pewne fragmenty mają być wykonywane w źródle danych, ich przetwarzanie jest przekazywane do realizacji zdalnej. Po przetworzeniu fragmentów przez źródło danych wyniki są pobierane i zwracane do programu DB2 UDB. Jeśli część przetwarzania była realizowana przez program DB2 UDB, wyniki uzyskane lokalnie są łączone z wynikami pobranymi ze źródła danych. W końcowym etapie program DB2 UDB zwraca wszystkie wyniki klientowi.

Pojęcia pokrewne:

- “Kompilator SQL” na stronie 8
- “Kompensacja” na stronie 9
- “Strojenie przetwarzania zapytań” na stronie 133

Kompensacja

Serwer stowarzyszony DB2[®] nie przekazuje fragmentów zapytania do źródeł danych, jeśli te nie mogą ich wykonać, lub jeśli serwer stowarzyszony może je wykonać szybciej niż źródło danych. Dla przykładu założmy, że dialekt SQL używany w źródle danych nie obsługuje grupowania CUBE w klauzuli GROUP BY. Serwer stowarzyszony otrzymuje zapytanie zawierające grupowanie typu CUBE i odwołujące się do tabeli w tym źródle danych. Program DB2 Information Integrator nie przekaże przetwarzania grupowania CUBE do źródła danych, lecz przetworzy je samodzielnie. Możliwość przetwarzania przez program DB2 Information Integrator instrukcji SQL, które nie są obsługiwane przez źródło danych, nazywa się *kompensacją*.

Serwer stowarzyszony kompensuje niedostatki funkcjonalności źródła danych na dwa sposoby:

- Może zlecić, by w źródle danych wykonane zostały operacje równoważne funkcjonalnie operacjom DB2 wywołanym w zapytaniu. Załóżmy, że w źródle danych nie jest zaimplementowana funkcja kotangens ($COT(x)$), ale jest dostępna funkcja tangens ($TAN(x)$). Program DB2 Information Integrator może zlecić wykonanie w źródle danych obliczeń ($1/TAN(x)$), które dają wynik równoważny funkcji ($COT(x)$).
- Może sprowadzić zestaw danych na serwer stowarzyszony i wykonać funkcję lokalnie.

Dla relacyjnych źródeł danych każdy typ systemu RDBMS obsługuje pewien podzbiór międzynarodowego standardu SQL. Oprócz tego w niektórych systemach obsługiwane są konstrukcje SQL wykraczające poza ten standard. *Dialekt SQL* jest to zbiór zasad i elementów języka SQL obsługiwanych przez dany typ systemu zarządzania relacyjnymi bazami danych. Jeśli dana konstrukcja SQL należy do dialektu SQL programu DB2, ale nie do dialektu relacyjnego źródła danych, serwer stowarzyszony może zaimplementować tę konstrukcję zamiast źródła danych.

Program DB2 Information Integrator może kompensować różnice między dialektami SQL. Przykładem takiego działania jest klauzula wspólnego wyrażenia tabelowego. Dialekt SQL używany w programie DB2 obejmuje klauzulę wspólnego wyrażenia tabelowego. W klauzuli takiej można podać nazwę, za pośrednictwem której wszystkie klauzule FROM w wyrażeniu pełnej selekcji mogą odwoływać się do tabeli wynikowej. Serwer stowarzyszony realizuje zapytania SQL ze wspólnym wyrażeniem tabelowym względem źródła danych, nawet jeśli dialekt SQL źródła nie obsługuje takich wyrażań.

Dzięki mechanizmowi kompensacji serwer stowarzyszony oferuje pełną obsługę dialektu SQL programu DB2 w zapytaniach kierowanych do źródeł danych. Daje to ewidentne korzyści w przypadku źródeł danych obsługujących SQL w ograniczonym zakresie lub nieobsługujących go wcale. Jednak zapytania kierowane do systemu stowarzyszonego należy formułować w dialekcie SQL programu DB2, chyba że praca odbywa się w ramach sesji tranzytowej.

Pojęcia pokrewne:

- “Sesje tranzytowe” na stronie 10

Sesje tranzytowe

Korzystając ze specjalnego trybu działania, zwanego *tranzytem*, można wprowadzać instrukcje SQL bezpośrednio do źródeł danych. Instrukcje SQL muszą spełniać wymogi dialektu SQL używanego w odpowiednim źródle danych. Sesja tranzytowa pozwala wykonać operacje, których wykonanie nie jest możliwe za pośrednictwem interfejsu i języka SQL programu DB2[®]. Na przykład w ramach sesji tranzytowej można utworzyć procedurę, indeks lub wykonywać zapytania w rodzimym dialekcie źródła danych.

Obecnie możliwa jest realizacja wyłącznie sesji tranzytowych w języku SQL. Możliwe, że w przyszłości źródła danych będą obsługiwały w ramach sesji tranzytowej także języki inne niż SQL.

Sesje tranzytowe mogą także posłużyć do wykonywania działań, które nie są obsługiwane w języku SQL, na przykład niektórych zadań administracyjnych. Nie można jednak w ramach sesji tranzytowych wykonywać wszystkich zadań administracyjnych. Na przykład jest możliwe tworzenie i usuwanie tabel w źródle danych, ale nie jest możliwe uruchamianie lub zatrzymywanie zdalnej bazy danych.

W ramach sesji tranzytowej można korzystać zarówno ze statycznego, jak i dynamicznego SQL.

Do sterowania sesjami tranzytowymi serwer stowarzyszony udostępnia następujące instrukcje SQL:

SET PASSTHRU

Otwiera sesję tranzytową. Wydanie kolejnej instrukcji SET PASSTHRU w celu zapoczątkowania nowej sesji tranzytowej powoduje zamknięcie dotychczasowej sesji tranzytowej.

SET PASSTHRU RESET

Powoduje zakończenie bieżącej sesji tranzytowej.

GRANT (uprawnienia serwera)

Nadaje użytkownikowi, grupie, liście identyfikatorów autoryzowanych użytkowników lub kategorii PUBLIC prawo do inicjowania sesji tranzytowych z określonym źródłem danych.

REVOKE (uprawnienia serwera)

Odbiera uprawnienie do inicjowania sesji tranzytowych.

Sesje tranzytowe podlegają następującym ograniczeniom:

- Należy posługiwać się dialektem SQL i komendami rozpoznawanymi przez źródło danych — nie można korzystać z dialektu SQL programu DB2. W efekcie w zapytaniach nie należy powoływać pseudonimu, lecz bezpośrednio obiekty źródła danych.
- Wykonując operacje UPDATE lub DELETE w ramach sesji tranzytowej, nie można korzystać z warunku WHERE CURRENT OF CURSOR.
- Sesje tranzytowe nie umożliwiają operowania na obiektach LOB.

Pojęcia pokrewne:

- “Opakowania i moduły opakowujące” na stronie 11
- “Bezpośrednie kierowanie zapytań do źródeł danych w sesji tranzytowej” na stronie 213

Opakowania i moduły opakowujące

Opakowania są to mechanizmy, za pośrednictwem których serwer stowarzyszony oddziałuje ze źródłami danych. Serwer stowarzyszony korzysta z procedur zapisanych w bibliotece nazywanej *modułem opakowującym*, która stanowi implementację opakowania. Procedury te umożliwiają serwerowi stowarzyszonemu wykonywanie takich operacji, jak łączenie się ze źródłem danych i iteracyjne odczytywanie z niego danych. Do zarejestrowania opakowania w stowarzyszonej bazie danych właściciel stowarzyszonej instancji DB2® używa zwykle instrukcji CREATE WRAPPER. Opakowanie można zarejestrować jako chronione lub zaufane za pomocą opcji opakowania DB2_FENCED.

Należy utworzyć po jednym opakowaniu na każdy typ źródła danych, do którego wymagany jest dostęp. Dla przykładu założmy, że wymagany jest dostęp do trzech tabel programu DB2 for z/OS™, jednej tabeli programu DB2 for iSeries™, dwóch tabel programu Informix® oraz jednego widoku programu Informix. Wymagane jest utworzenie jednego opakowania dla obiektów źródła danych DB2 i jednego dla obiektów źródła danych Informix. Po zarejestrowaniu tych opakowań w stowarzyszonej bazie danych można ich używać jako drogi dostępu do wszystkich obiektów w odpowiednim źródle danych. Na przykład opakowanie DRDA® umożliwia dostęp do wszystkich obiektów źródeł danych z rodziny DB2, czyli DB2 dla systemów Linux, UNIX® i Windows®, DB2 for z/OS and OS/390®, DB2 for iSeries oraz DB2 Server for VM and VSE.

Za pośrednictwem definicji serwera i pseudonimu identyfikowane są dane specyficzne (nazwa, lokalizacja itd.) każdego z obiektów źródła danych.

Opakowanie realizuje wiele zadań. Do zadań tych należy między innymi:

- Nawiązywanie połączeń ze źródłami danych. Opakowanie korzysta ze standardowego połączeniowego interfejsu API dla źródła danych.
- Wprowadzanie zapytań do źródła danych.
 - W przypadku źródeł danych obsługujących język SQL zapytanie jest wprowadzane w tym języku.
 - W przypadku źródeł danych nieobsługujących języka SQL zapytanie jest tłumaczone na rodzimy język zapytań źródła danych lub na sekwencję wywołań funkcji API.
- Odbieranie tabel wynikowych ze źródła danych. Odbierając wyniki, opakowanie korzysta ze standardowego interfejsu API źródła danych.
- Odpowiadanie na zapytania serwera stowarzyszonego dotyczące domyślnych odwzorowań typów danych dla źródła danych. Opakowanie zawiera odwzorowania domyślne, używane przy tworzeniu pseudonimów dla obiektów źródła danych. W przypadku opakowań relacyjnych odwzorowania typów tworzone przez użytkownika zastępują odwzorowania domyślne. Odwzorowania typów utworzone przez użytkownika są przechowywane w katalogu globalnym.
- Odpowiadanie na zapytania serwera stowarzyszonego dotyczące domyślnych odwzorowań funkcji dla źródła danych. Opakowanie zawiera informacje, na podstawie których serwer stowarzyszony ustala, czy i w jaki sposób funkcje DB2 są odwzorowane do funkcji źródła danych. Z informacji tych korzysta kompilator SQL, decydując, czy źródło danych może wykonać operacje związane z realizacją zapytania. W przypadku opakowań relacyjnych odwzorowania funkcji tworzone przez użytkownika zastępują odwzorowania domyślne. Odwzorowania funkcji zdefiniowanej przez użytkownika są przechowywane w katalogu globalnym.

Opcje opakowań są używane do konfigurowania opakowań lub określania sposobu, w jaki program DB2 Information Integrator korzysta z opakowań.

Pojęcia pokrewne:

- “Definicje serwerów i opcje serwera” na stronie 13

Zadania pokrewne:

- “Trusted and fenced mode process environments” w podręczniku *IBM DB2 Information Integrator Wrapper Developer’s Guide*
- “Modyfikowanie opakowania” na stronie 27

Informacje pokrewne:

- “ALTER WRAPPER statement” w podręczniku *SQL Reference, Volume 2*
- Rozdział 20, “Opcje opakowań w systemie stowarzyszonym”, na stronie 223
- “Domyślne nazwy opakowań” na stronie 12

Domyślne nazwy opakowań

Dla każdego obsługiwanego źródła danych istnieją osobne opakowania. Niektóre z opakowań mają domyślne nazwy. Jeśli przy tworzeniu opakowania wybrana zostanie nazwa domyślna, serwer stowarzyszony automatycznie dobierze bibliotekę źródła danych skojarzoną z tym opakowaniem.

Tabela 2. Domyślne nazwy opakowań dla poszczególnych źródeł danych.

Źródło danych	Domyślne nazwy opakowań
DB2 Universal Database™ dla systemów Linux, UNIX i Windows®	DRDA
DB2 Universal Database for z/OS and OS/390®	DRDA
DB2 Universal Database for iSeries,	DRDA
DB2 Server for VM and VSE	DRDA
Informix	INFORMIX
Microsoft® SQL Server	MSSQLODBC3
ODBC	ODBC
OLE DB	OLEDB
Oracle	NET8
Sybase	CTLIB
Teradata	TERADATA
BLAST	Brak
BioRS	Brak
Documentum	Brak
Entrez	Brak
Extended Search	Brak
HMMER	Brak
Microsoft Excel	Brak
Pliki o strukturze tabeli	Brak
Usługi WWW	Brak
WebSphere Business Integration	Brak
XML	Brak

Pojęcia pokrewne:

- “Opakowania i moduły opakowujące” na stronie 11

Definicje serwerów i opcje serwera

Po utworzeniu opakowań dla źródeł danych właściciel instancji systemu stowarzyszonego definiuje źródła danych w stowarzyszonej bazie danych. W tym celu właściciel instancji określa nazwę źródła danych oraz inne stosowne informacje. Informacje te obejmują:

- typ i wersję źródła danych,
- nazwę bazy danych będącej źródłem danych (tylko system RDBMS),
- metadane właściwe dla źródła danych.

Na przykład źródło danych z rodziny DB2® może zawierać więcej niż jedną bazę danych. Definicja musi zawierać informację o tym, z którą bazą danych serwer stowarzyszony może nawiązywać połączenia. Natomiast źródło danych Oracle udostępnia tylko jedną bazę danych, dlatego serwer stowarzyszony może nawiązywać połączenia z taką bazą danych, nie znając jej nazwy. W związku z tym nazwa bazy danych nie jest uwzględniana w definicji źródła danych Oracle.

Nazwa oraz inne informacje podawane przez właściciela instancji na serwerze stowarzyszonym są łącznie nazywane *definicją serwera*. Źródła danych realizują kierowane do nich żądania dostępu do danych, pełniąc tym samym rolę serwerów.

Instrukcje CREATE SERVER i ALTER SERVER są używane do tworzenia i modyfikowania definicji serwerów.

Niektóre z informacji zawartych w definicji serwera przechowywane są w postaci *opcji serwera*. Przy tworzeniu definicji serwera duże znaczenie ma właściwe operowanie możliwymi do określenia opcjami serwera. Niektóre z opcji serwera służą do konfigurowania opakowania, inne wpływają na sposób, w jaki program DB2 Information Integrator współpracuje z opakowaniem.

Opcje serwera mogą być skonfigurowane w sposób zapewniający zachowywanie ich między kolejnymi połączeniami ze źródłem danych lub ich ustawianie tylko na czas pojedynczego połączenia.

Pojęcia pokrewne:

- “Odwzorowania użytkowników” na stronie 14

Informacje pokrewne:

- Rozdział 21, “Opcje serwera w systemie stowarzyszonym”, na stronie 225

Odwzorowania użytkowników

Gdy serwer stowarzyszony przekazuje żądanie do przetworzenia w źródle danych, w pierwszej kolejności konieczne jest nawiązanie połączenia ze źródłem danych.

W przypadku źródeł danych serwer stowarzyszony wykonuje to zadanie, używając względem określonego źródła danych poprawnego identyfikatora użytkownika i hasła. Jeśli do połączenia ze źródłem danych wymagany jest identyfikator i hasło użytkownika, można zdefiniować skojarzenie między identyfikatorem autoryzowanego użytkownika na serwerze stowarzyszonym a identyfikatorem i hasłem zdefiniowanym w źródle danych. To skojarzenie może zostać utworzone dla każdego identyfikatora użytkownika, który wysyła żądania rozproszone przy użyciu systemu stowarzyszonego. Skojarzenie takie jest nazywane *odwzorowaniem użytkownika*.

Gdy identyfikator i hasło użytkownika używane do połączenia ze stowarzyszoną bazą danych są takie same, jak używane podczas korzystania ze zdalnego źródła danych, nie ma potrzeby tworzenia odwzorowania użytkownika.

Zadania pokrewne:

- “Registering user mappings for a data source” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Informacje pokrewne:

- Rozdział 22, “Opcje odwzorowań użytkowników w systemach stowarzyszonych”, na stronie 241

Pseudonimy i obiekty źródła danych

Po utworzeniu definicji serwera i odwzorowań użytkowników właściciel instancji systemu stowarzyszonego tworzy pseudonimy. *Pseudonim* jest to identyfikator używany w celu odwoływania się do obiektu rezydującego w źródle danych. Obiekty, do którego odwołują się pseudonimy, nazywane są *obiettami źródła danych*.

Pseudonimy, w odróżnieniu od aliasów, nie są alternatywnymi nazwami obiektów źródła danych. Są one wskaźnikami, za pośrednictwem których serwer stowarzyszony odwołuje się do obiektów. Do definiowania pseudonimów używa się najczęściej instrukcji CREATE NICKNAME wraz z konkretnymi opcjami kolumn pseudonimu i opcjami pseudonimu.

Gdy użytkownik końcowy lub aplikacja kliencka wprowadza do serwera stowarzyszonego żądanie rozproszone, nie musi w tym żądaniu definiować źródeł danych. Zamiast tego żądanie zawiera odwołania do obiektów źródła danych za pośrednictwem ich pseudonimów. Pseudonimy są odwzorowane na określone obiekty w źródle danych. Odwzorowania te eliminują potrzebę dodawania do pseudonimów kwalifikatorów w postaci nazw źródeł danych. Rzeczywista lokalizacja obiektów w źródłach danych jest całkowicie przezroczysta dla użytkowników i aplikacji klienckich.

Załóżmy, że zdefiniowano pseudonim *DEPT*, który ma reprezentować tabelę w bazie danych Informix[®] noszącą nazwę *NFX1.PERSON*. Na serwerze stowarzyszonym dopuszczalne jest użycie instrukcji SQL SELECT * FROM *DEPT*. Jednak użycie instrukcji SELECT * FROM *NFX1.PERSON* na serwerze stowarzyszonym jest niedozwolone (z wyjątkiem sesji tranzytowych), chyba że na tym serwerze znajduje się tabela lokalna *NFX1.PERSON*.

Podczas tworzenia pseudonimu dla obiektu źródła danych do katalogu globalnego są wprowadzane metadane opisujące ten obiekt. Na podstawie tych metadanych oraz informacji z opakowania optymalizator zapytań rozpatruje sposoby usprawnienia dostępu do danego obiektu. Na przykład, jeśli pseudonim dotyczy tabeli mającej indeks, informacje o tym indeksie figurują w katalogu globalnym. Opakowanie zawiera odwzorowania między typami danych programu DB2[®] a typami danych w źródle danych.

W bieżącej wersji nie można wykonywać niektórych operacji na pseudonimach przy użyciu programów narzędziowych DB2 UDB.

Nie można użyć programu narzędziowego Cross Loader do ładowania danych z innych systemów relacyjnych baz danych do pseudonimu.

Pojęcia pokrewne:

- “Opcje kolumny pseudonimu” na stronie 17

Informacje pokrewne:

- Rozdział 24, “Opcje kolumn pseudonimu dla systemów stowarzyszonych”, na stronie 253
- Rozdział 23, “Opcje pseudonimu dla systemów stowarzyszonych”, na stronie 243
- “Poprawne obiekty źródła danych” na stronie 16

Poprawne obiekty źródła danych

Pseudonimy identyfikują obiekty źródeł danych, do których odwołuje się użytkownik. W poniższej tabeli przedstawiono wykaz typów obiektów, dla których można utworzyć pseudonimy w stowarzyszonej bazie danych.

Tabela 3. Poprawne obiekty źródła danych

Źródło danych	Poprawne obiekty
DB2 dla systemów Linux, UNIX i Windows	pseudonimy, zmaterializowane tabele zapytań, tabele i widoki
DB2 for z/OS and OS/390	tabele, widoki
DB2 for iSeries	tabele, widoki
DB2 for VM and VSE	tabele, widoki
Informix	tabele, widoki, synonimy
Microsoft SQL Server	tabele, widoki
ODBC	tabele, widoki
Oracle	tabele, widoki, synonimy
Sybase	tabele, widoki
Teradata	tabele, widoki
BLAST	pliki FASTA poindeksowane na potrzeby algorytmów wyszukiwania BLAST
BioRS	Banki danych BioRS
Documentum	obiekty i tabele zarejestrowane w bazie Documentum Docbase
Entrez	bazy danych Entrez
Extended Search	pliki z takich źródeł danych, jak bazy danych Lotus Notes, Microsoft Access, Microsoft Index Server, wyszukiwarki internetowe i katalogi LDAP
HMMER	pliki baz danych HMM (biblioteki hierarchicznych modeli Markowa, na przykład PFAM), przeszukiwane za pomocą programów HMMER hmmpfam lub hmmsearch.
Microsoft Excel	pliki .xls (dostępny jest tylko pierwszy arkusz w skoroszycie)
Pliki o strukturze tabeli	Pliki tekstowe o określonym formacie.
Adaptery Websphere Business Integration	Obiekty biznesowe Websphere Business Integration odwzorowane na interfejsy BAPI w systemie SAP, komponenty biznesowe w systemie Siebel i interfejsy komponentów w systemie PeopleSoft
Usługi WWW	Operacje w pliku języka opisu usług WWW
pliki znaczników w formacie XML	zbiory elementów w dokumencie XML

Pojęcia pokrewne:

- “Pseudonimy i obiekty źródła danych” na stronie 15
- “Opcje kolumny pseudonimu” na stronie 17

Opcje kolumny pseudonimu

Do katalogu globalnego można samodzielnie wprowadzić dodatkowe informacje na temat obiektu, którego dotyczy pseudonim. Takie metadane są opisem wartości w niektórych kolumnach obiektu źródła danych. Metadane te są skojarzone z parametrami nazywanymi *opcjami kolumn pseudonimu*. Opcje kolumn pseudonimu informują opakowanie o konieczności traktowania danych w kolumnie w szczególny sposób. Kierując się informacjami zawartymi w metadanych, kompilator SQL i optymalizator zapytań opracowują doskonalsze plany dostępu do danych.

Opcje kolumn pseudonimu umożliwiają także przekazywanie do opakowania innych informacji. Na przykład w przypadku źródeł danych XML opcja kolumny pseudonimu przekazuje do opakowania wyrażenie XPath używane przez opakowanie podczas wyodrębniania danej kolumny z dokumentu XML.

W systemie stowarzyszonym serwer DB2[®] traktuje obiekt źródła danych, do którego odwołuje się pseudonim, tak jak gdyby był on lokalną tabelą DB2. Dzięki temu można ustawić opcje kolumn pseudonimu dla każdego obiektu źródła danych, dla którego zdefiniowano pseudonim. Niektóre z opcji kolumn pseudonimu są przeznaczone dla ściśle określonych typów źródeł danych i mogą być stosowane tylko w odniesieniu do tych źródeł.

Założmy, że w źródle danych ustawiona jest inna kolejność zestawiania niż w stowarzyszonej bazie danych. Serwer stowarzyszony w takiej sytuacji z reguły nie przekazywałby wykonania operacji sortowania kolumn z danymi znakowymi do źródła danych. Dane byłyby przekazywane do stowarzyszonej bazy danych i sortowane lokalnie. Założmy jednak, że kolumna ma przypisany typ znakowy (CHAR lub VARCHAR), mimo że zawiera wyłącznie dane złożone z cyfr ('0', '1', ..., '9'). Informację o tym można przekazać, przypisując opcji kolumny pseudonimu NUMERIC_STRING wartość 'Y'. Dzięki temu optymalizator zapytań DB2 uzyskuje możliwość realizacji sortowania w źródle danych. Wykonując sortowanie zdalnie, można uniknąć narzutu związanego z przekazywaniem danych na serwer stowarzyszony i wykonaniem sortowania lokalnie.

Dla pseudonimów relacyjnych opcje kolumn można zdefiniować przy użyciu instrukcji ALTER NICKNAME. Dla pseudonimów nierelacyjnych opcje kolumn definiuje się za pomocą instrukcji CREATE NICKNAME i ALTER NICKNAME.

Pojęcia pokrewne:

- “Odwzorowania typów danych” na stronie 17

Zadania pokrewne:

- “Praca z pseudonimami” na stronie 100

Informacje pokrewne:

- Rozdział 24, “Opcje kolumn pseudonimu dla systemów stowarzyszonych”, na stronie 253

Odwzorowania typów danych

Typy danych w źródle danych muszą być odwzorowane na odpowiednie typy danych programu DB2[®], aby serwer stowarzyszony mógł pobierać dane z tych źródeł. Oto niektóre przykłady domyślnych odwzorowań typów danych:

- typ Oracle FLOAT odwzorowany jest na typ DB2 DOUBLE
- typ Oracle DATE odwzorowany jest na typ DB2 TIMESTAMP
- typ DB2 dla z/OS[™] DATE odwzorowany jest na typ DB2 DATE.

W przypadku większości źródeł danych domyślne odwzorowania typów są zdefiniowane w opakowaniach. Domyślne odwzorowania typów dla źródeł danych z rodziny DB2 określone są w opakowaniu DRDA[®]. Domyślne odwzorowania typów dla źródeł Informix[®] zawiera opakowanie INFORMIX, itd.

W przypadku niektórych nierelacyjnych źródeł danych wymagane jest podanie informacji o typie danych w instrukcji CREATE NICKNAME. Podczas tworzenia pseudonimu dla każdej kolumny obiektu w źródle danych należy zdefiniować odpowiedni typ danych programu DB2 dla systemów Linux, UNIX[®] i Windows[®]. Każda kolumna musi być odwzorowana na określone pole lub kolumnę w obiekcie źródła danych.

W przypadku relacyjnych źródeł danych możliwe jest zastąpienie domyślnych odwzorowań typów. Na przykład typ danych Informix INTEGER jest domyślnie odwzorowywany na typ DB2 INTEGER. Można jednak przesłonić odwzorowania domyślne i odwzorować typ Informix INTEGER na typ DB2 DECIMAL(10,0).

Tworzenie nowych odwzorowań typów oraz modyfikowanie odwzorowań domyślnych powinno się odbywać przed utworzeniem pseudonimów. W przeciwnym razie pseudonimy utworzone przed zmianą odwzorowań nie będą odzwierciedlały zmian.

Pojęcia pokrewne:

- “Odwzorowania typów danych w systemie stowarzyszonym” na stronie 47

Odwzorowania funkcji

Aby serwer stowarzyszony mógł rozpoznać funkcję źródła danych, funkcja ta musi zostać odwzorowana na istniejącą funkcję programu DB2[®] dla systemów Linux, UNIX[®] i Windows[®]. W programie DB2 Information Integrator zdefiniowane są domyślne odwzorowania między istniejącymi wbudowanymi funkcjami źródła danych a wbudowanymi funkcjami programu DB2. W przypadku większości źródeł danych domyślne odwzorowania funkcji są zdefiniowane w opakowaniach. Domyślne odwzorowania na funkcję programu DB2 for z/OS[™] and OS/390[®] zdefiniowane są w opakowaniu DRDA[®]. Domyślne odwzorowania na funkcje Sybase są zapisane w opakowaniach CTLIB i DBLIB itd.

W przypadku relacyjnych źródeł danych można utworzyć odwzorowanie funkcji, gdy zachodzi potrzeba użycia funkcji źródła danych nierozpoznawanej przez serwer stowarzyszony. Tworzone odwzorowanie kojarzy funkcję źródła danych z odpowiadającą jej funkcją DB2 w stowarzyszonej bazie danych. Odwzorowania funkcji są często używane, gdy w źródle danych udostępniona zostaje nowa funkcja wbudowana lub zdefiniowana przez użytkownika. Odwzorowania funkcji pozwalają także wybrnąć z sytuacji, gdy odpowiednia funkcja programu DB2 nie istnieje. Konieczne jest wówczas utworzenie dodatkowo szablonu funkcji.

Pojęcia pokrewne:

- “Odwzorowania funkcji w systemie stowarzyszonym” na stronie 59
- “Specyfikacje indeksu” na stronie 19

Specyfikacje indeksu

Podczas tworzenia pseudonimu dla tabeli źródła danych informacje o indeksach związanych z tą tabelą są dodawane do katalogu globalnego. Optymalizator zapytań wykorzystuje te informacje do przyspieszenia przetwarzania żądań rozproszonych. Informacje na temat indeksu w źródle danych są zapisane w katalogu w postaci zestawu metadanych nazywanego *specyfikacją indeksu*. Serwer stowarzyszony nie tworzy specyfikacji indeksu przy tworzeniu pseudonimu dla:

- tabel bez indeksów;
- widoków, którym w katalogu zdalnym z reguły nie są przypisane żadne informacje indeksowe;
- obiektów w źródle danych pozbawionym zdalnego katalogu, z którego serwer stowarzyszony mógłby uzyskać informacje o indeksach.

Załóżmy, że tabela zostaje uzupełniona o nowy indeks, dodany do tych, które były zdefiniowane w momencie tworzenia pseudonimu. Ponieważ dane o indeksach są przekazywane do katalogu globalnego tylko w chwili tworzenia pseudonimu, serwer stowarzyszony nie jest poinformowany o istnieniu nowego indeksu. Podobnie podczas tworzenia pseudonimu dla widoku serwer stowarzyszony nie dysponuje żadnymi informacjami na temat tabeli (oraz jej indeksów), na bazie której widok został wygenerowany. W takiej sytuacji można samodzielnie wpisać potrzebne informacje do katalogu globalnego. Można utworzyć specyfikację indeksu dla tabel pozbawionych indeksu. Specyfikacja indeksu informuje optymalizator zapytań, które kolumny w tabeli należy przeszukać w celu najszybszego odnalezienia danych.

Pojęcia pokrewne:

- “Specyfikacje indeksów w systemie stowarzyszonym” na stronie 73

Kolejności zestawiania

Kolejność, w jakiej dane znakowe są sortowane w bazie danych, zależy od struktury danych i od kolejności zestawiania zdefiniowanej w bazie danych.

Załóżmy, że dane w bazie danych są zapisane w postaci wyłącznie wielkich liter, bez cyfr i znaków specjalnych. Sortowanie danych powinno dać ten sam efekt bez względu na to, czy dane są sortowane w źródle danych, czy w stowarzyszonej bazie danych. Różnica kolejności zestawiania między serwerami nie powinna mieć znaczenia dla wyników sortowania. Także wtedy, gdy dane mają postać wyłącznie małych liter lub wyłącznie cyfr, ich sortowanie powinno dać taki sam wynik niezależnie od tego, gdzie jest wykonywane.

Jeśli dane mają jedną z poniższych postaci:

- kombinacje liter i cyfr;
- litery małe i wielkie;
- znaki specjalne, jak @, #, €;

sortowanie tych danych może dać odmienne efekty, jeśli kolejności zestawiania w stowarzyszonej bazie danych i w źródle danych są różne.

Ogólne mówiąc, *kolejność zestawiania* jest to zdefiniowana metoda porządkowania danych znakowych, określająca, czy dany znak przypada przed, po, czy na równi z innym znakiem.

Jak kolejność zestawiania wpływa na porządek sortowania

Kolejność zestawiania określa porządek sortowania znaków w ramach kodowego zestawu znaków. *Zestaw znaków* jest to zbiór znaków używanych w systemie komputerowym lub języku programowania. W *kodowanym* zestawie znaków każdemu znakowi przypisany jest osobny numer z zakresu od 0 do 255 (lub jego szesnastkowy odpowiednik). Liczby te są nazywane *punktami kodowymi*; ogół przypisań liczb do znaków w zestawie nosi nazwę *strony kodowej*.

Oprócz przypisania do znaku, punkt kodowy może być odwzorowany na pozycję znaku w ramach kolejności sortowania. Z punktu widzenia technicznego kolejność zestawiania jest zbiorowym odwzorowaniem punktów kodowych zestawu znaków na pozycję w ramach kolejności sortowania danego zestawu znaków. Pozycja znaku jest reprezentowana przez liczbę, która jest nazywana *wagą* znaku. W najprostszej kolejności zestawiania, nazywanej *kolejnością tożsamościową*, wagi są identyczne z punktami kodowymi.

Załóżmy, że baza danych ALPHA korzysta z domyślnej kolejności zestawiania strony kodowej EBCDIC, a w bazie danych BETA używana jest domyślna kolejność zestawiania strony kodowej ASCII. Kolejności sortowania obu tych baz byłyby odmienne, co obrazuje poniższy przykład:

```
SELECT.....
```

```
ORDER BY KOL2
```

Sortowanie EBCDIC

Sortowanie ASCII

KOL2

KOL2

V1G

7AB

Y2W

V1G

7AB

Y2W

Podobnie porównywanie znaków w bazie danych zależy od kolejności zestawiania zdefiniowanej dla tej bazy danych. W tym przykładzie baza danych ALPHA korzysta z domyślnej kolejności zestawiania strony kodowej EBCDIC. Baza danych BETA korzysta z domyślnej kolejności zestawiania strony kodowej ASCII. Porównywanie znaków obu tych baz danych dałoby odmienne wyniki, co obrazuje poniższy przykład:

```
SELECT.....
```

```
WHERE KOL2 > 'TT3'
```

Wyniki dla EBCDIC

Wyniki dla ASCII

KOL2

KOL2

TW4

TW4

X82

X82

39G

Konfigurowanie lokalnej kolejności zestawiania w celu optymalizacji zapytań

Administratorzy mogą tworzyć stowarzyszone bazy danych o określonej kolejności zestawiania, dopasowanej do charakterystyki źródła danych. Następnie dla każdej definicji źródła danych opcja serwera `COLLATING_SEQUENCE` jest ustawiana na 'Y'. Ustawienie to poświadcza dopasowanie kolejności zestawiania między serwerem stowarzyszonym a źródłem danych.

Kolejność zestawiania stowarzyszonej bazy danych jest konfigurowana przy użyciu instrukcji `CREATE DATABASE`. Za jej pośrednictwem można wybrać jedną z następujących kolejności:

- kolejność tożsamościową;
- kolejność *systemową* (kolejność używaną w systemie operacyjnym, pod kontrolą którego działa baza danych);
- kolejność *niestandardową* (kolejność zestawiania predefiniowaną w programie DB2 UDB lub zdefiniowaną przez użytkownika).

Założmy, że źródłem danych jest program DB2 for z/OS and OS/390. Sortowanie definiowane klauzulą ORDER BY jest realizowane w ramach kolejności zestawiania na bazie strony kodowej EBCDIC. Aby pobrać dane ze źródła DB2 for z/OS and OS/390 posortowane zgodnie z klauzulą ORDER BY, należy skonfigurować odpowiednio stowarzyszoną bazę danych, ustawiając predefiniowaną kolejność zestawiania na podstawie właściwej strony kodowej EBCDIC.

Pojęcia pokrewne:

- “Cechy serwera wpływające na możliwości przekazywania” na stronie 136
- “Collating sequences in a federated system” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Zadania pokrewne:

- “Creating a federated database” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Informacje pokrewne:

- “National language versions” w podręczniku *Administration Guide: Planning*
- “Federated database national language considerations” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Jak przebiega interakcja z systemem stowarzyszonym

Ponieważ stowarzyszona baza danych jest instancją DB2[®] Universal Database, do obsługi systemu stowarzyszonego może służyć dowolne z poniższych narzędzi:

- procesor wiersza komend DB2 (CLP)
- interfejs graficzny Centrum komend DB2
- interfejs graficzny Centrum sterowania DB2
- aplikacje
- narzędzia z rodziny DB2
- dostawcy usług WWW

W dokumentacji systemu stowarzyszonego znajdują się opisy komend i instrukcji SQL, które mogą być wprowadzane za pośrednictwem procesora wiersza komend DB2 lub graficznego interfejsu Centrum komend DB2. W dokumentacji zaznaczono też, które zadania można wykonywać przy użyciu interfejsu graficznego Centrum sterowania DB2. Ponieważ obsługa interfejsu Centrum sterowania DB2 odbywa się w sposób intuicyjny, w niniejszym dokumencie pominięto opis wykonywania odpowiednich zadań w Centrum sterowania.

Procesor wiersza komend DB2 (CLP)

Większość czynności wymaganych w ramach procesów instalacji, konfiguracji, strojenia i konserwacji systemu stowarzyszonego można wykonać za pośrednictwem procesora wiersza komend DB2. W niektórych przypadkach konieczne jest użycie procesora wiersza komend DB2 lub Centrum komend DB2. Do zadań tych należy:

- tworzenie, modyfikacja lub usuwanie odwzorowań typów zdefiniowanych przez użytkownika;
- tworzenie, modyfikacja lub usuwanie odwzorowań funkcji zdefiniowanej przez użytkownika.

Centrum komend DB2

Za pośrednictwem Centrum komend DB2 można tworzyć i uruchamiać żądania rozproszone bez potrzeby samodzielnego wpisywania rozbudowanych instrukcji SQL. Ponadto Centrum komend DB2 jest używane przy dostrajaniu wydajności systemu stowarzyszonego. Za pośrednictwem Centrum komend DB2 można wygodnie korzystać z funkcji wyjaśniania, która pozwala przeglądać plany dostępu opracowane dla żądań rozproszonych. Centrum komend DB2 umożliwia ponadto współpracę z narzędziem Asysta SQL.

Centrum sterowania DB2

Interfejs graficzny Centrum sterowania DB2 pozwala wykonywać większość zadań w ramach procesu instalacji, konfiguracji i modyfikacji systemu stowarzyszonego. Interfejs Centrum sterowania zbudowany jest w oparciu o panele, czyli okna dialogowe i ekrany kreatorów, które pomagają w krokowym wykonaniu zadania. Panele te zawierają interakcyjne wskazówki, wyświetlane po zatrzymaniu wskaźnika myszy nad polem sterującym, takim jak lista lub przycisk polecenia. Ponadto każdy z paneli wyposażony jest w przycisk pomocy, powodujący wyświetlenie informacji o przeznaczeniu danego panelu oraz odsyłaczy do pokrewnych tematów i informacji uzupełniających.

Do utworzenia obiektów stowarzyszonych można użyć kreatora, można też tworzyć te obiekty pojedynczo.

W celu skonfigurowania dostępu do usług WWW oraz źródeł danych WebSphere® Business Integration i XML należy użyć Centrum sterowania DB2. Opcje wbudowane w Centrum sterowania DB2 ułatwiają wykonanie czynności wymaganych w celu skonfigurowania dostępu serwera stowarzyszonego do źródeł danych.

Interfejs graficzny Centrum sterowania DB2 jest najłatwiejszym narzędziem ułatwiającym wykonanie podstawowych czynności związanych z konfiguracją źródeł danych:

- tworzenia opakowań i konfigurowania opcji opakowań,
- określania zmiennych środowiskowych dla źródła danych,
- tworzenia definicji serwerów i konfigurowania opcji serwera,
- tworzenia odwzorowań użytkowników i konfigurowania opcji użytkowników,
- tworzenia pseudonimów oraz ustawiania opcji pseudonimów i opcji kolumn pseudonimów.

Po skonfigurowaniu dostępu serwera stowarzyszonego do źródeł danych można użyć Centrum sterowania DB2 do:

- modyfikowania konfiguracji źródła danych,
- monitorowania statusu pseudonimów i serwerów,
- aktualizowania statystyk dla pseudonimów,
- tworzenia i modyfikowania tabel buforowanych,
- określania ograniczeń informacyjnych dla pseudonimów,
- tworzenia zdalnych tabel w programie DB2 Information Integrator przy użyciu przezroczystego kodu DDL.

Aplikacje

Przy tworzeniu aplikacji współpracujących z bazą stowarzyszoną nie są wymagane żadne specjalne techniki programowania. Aplikacje łączą się z systemem w taki sam sposób, jak wszelkie inne aplikacje klienckie DB2. Aplikacje komunikują się ze stowarzyszoną bazą danych udostępnianą przez serwer stowarzyszony. W celu pobrania danych ze źródeł danych aplikacje wprowadzają zapytania SQL do stowarzyszonej bazy danych. Program DB2 Information Integrator następnie rozprawdza zapytania do odpowiednich źródeł danych, gromadzi pobrane dane i zwraca wynik do aplikacji. Ponieważ jednak program DB2 Information Integrator odwołuje się do źródeł danych za pośrednictwem pseudonimów, należy wziąć pod uwagę:

- ograniczenia języka SQL obowiązujące przy pracy z pseudonimami,
- sposoby wykonywania operacji na obiektach reprezentowanych przez pseudonimy.

Narzędzia z rodziny DB2

Obsługa stowarzyszonej bazy danych może się odbywać także za pośrednictwem narzędzi hosta lub warstwy pośredniej, takich jak:

- DB2 SPUFI na serwerze DB2 for z/OS™ and OS/390®
- Interactive SQL (STRSQL) na serwerze DB2 for iSeries™

Dostawcy usług WWW

Obsługa stowarzyszonej bazy danych może się także odbywać za pośrednictwem dostawców usług WWW przy użyciu opakowania dla usług WWW.

Pojęcia pokrewne:

- “The Web services wrapper and the Web services description language document” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Zadania pokrewne:

- “Adding Web services data sources to a federated server” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Część 2. Administrowanie i konserwacja

Rozdział 2. Modyfikowanie konfiguracji źródeł danych

Co pewien czas konieczne będzie wprowadzenie korekt w określonej na początku konfiguracji dostępu do źródeł danych. Na przykład w celu uzyskania dostępu do nowego serwera konieczne będzie zarejestrowanie definicji tego serwera. Po zmianie hasła do zdalnego źródła danych trzeba będzie zmienić odwzorowanie użytkownika między stowarzyszoną bazą danych a zdalnym źródłem danych. W celu poprawienia wydajności może być wymagane dodanie opcji kolumny do pseudonimu.

Ten rozdział zawiera:

- “Modyfikowanie opakowania”
- “Modyfikowanie definicji serwerów i opcji serwera” na stronie 28
- “Modyfikowanie odwzorowania użytkowników” na stronie 32
- “Modyfikowanie pseudonimu” na stronie 34
- “Usuwanie opakowania” na stronie 41
- “Usuwanie definicji serwera” na stronie 42
- “Usuwanie odwzorowania użytkowników” na stronie 43
- “Usuwanie pseudonimu” na stronie 44

Modyfikowanie opakowania

Po skonfigurowaniu opakowania można przystąpić do modyfikowania konfiguracji zgodnie z wymaganiami konkretnego systemu. Przy użyciu instrukcji ALTER WRAPPER można:

- dodawać, ustawiać lub usuwać opcje opakowania;
- ustawiać zmienne środowiskowe, rejestru lub profilu.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Ograniczenia:

Nie można usunąć opcji opakowania DB2_FENCED.

Procedura:

Opakowanie można zmodyfikować przy użyciu Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby zmodyfikować opakowanie z Centrum sterowania DB2:

1. Rozwiń folder Obiekty stowarzyszonej bazy danych. Obiekty opakowań zostaną wyświetlone w panelu zawartości, w oknie Centrum sterowania DB2.
2. Kliknij prawym przyciskiem myszy opakowanie, które chcesz zmienić, a następnie pozycję **Zmień** na liście działań. Zostanie otwarty notatnik Zmień opakowanie.
 - Wprowadź zmiany na stronie Ustawienia.
 - Kliknij przycisk **Ustaw zmienne**, aby ustawić zmienne środowiskowe źródła danych dla opakowania. Zmienne środowiskowe nie są wymagane dla wszystkich opakowań.

3. Kliknij przycisk **OK**, aby zmodyfikować opakowanie i zamknąć notatnik Zmień opakowanie.

Aby zmodyfikować opakowanie z wiersza komend DB2, użyj instrukcji ALTER WRAPPER.

Zadania pokrewne:

- “Sprawdzanie zmiennych środowiskowych dla źródeł danych” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*
- “Usuwanie opakowania” na stronie 41

Informacje pokrewne:

- “ALTER WRAPPER statement” w podręczniku *SQL Reference, Volume 2*
- “Modyfikowanie opakowania - przykłady” na stronie 28

Modyfikowanie opakowania - przykłady

Aby dla opakowania o nazwie drda zmienić wartość opcji DB2_FENCED na 'Y', należy użyć następującej instrukcji:

```
ALTER WRAPPER drda OPTIONS (SET DB2_FENCED 'Y');
```

Aby dla opakowania o nazwie odbc zmienić wartość opcji MODULE na '/opt/odbc/lib/libodbc.a(odbc.so)', należy użyć następującej instrukcji:

```
ALTER WRAPPER odbc OPTIONS (SET MODULE '/opt/odbc/lib/libodbc.a(odbc.so)');
```

Zadania pokrewne:

- “Modyfikowanie opakowania” na stronie 27

Modyfikowanie definicji serwerów i opcji serwera

Definicja serwera identyfikuje źródło danych na użytek stowarzyszonej bazy danych. W definicji serwera zawarta jest nazwa lokalna oraz inne informacje o serwerze źródła danych. Definicja serwera jest używana przez opakowanie, gdy do stowarzyszonej bazy danych wprowadzane są instrukcje SQL wykorzystujące pseudonimy. Do zmiany definicji serwera służy instrukcja ALTER SERVER.

Niektóre z informacji zawartych w definicji serwera przechowywane są w postaci opcji serwera. Podczas modyfikowania definicji serwera duże znaczenie ma właściwe operowanie dostępnymi opcjami serwera. Niektóre z opcji serwera służą do konfigurowania opakowania, inne wpływają na sposób, w jaki program DB2 UDB współpracuje z opakowaniem. Opcje serwera określa się jako parametry w instrukcjach CREATE SERVER i ALTER SERVER. Dodatkowo, dla relacyjnych źródeł danych, opcje serwera mogą być ustawiane tymczasowo za pomocą instrukcji SET SERVER OPTION. Wartość ustawiona w ten sposób zastępuje opcję serwera określoną w definicji serwera na czas pojedynczego połączenia ze stowarzyszoną bazą danych.

Słowo SERVER oraz nazwy parametrów zaczynające się od tego słowa w instrukcji ALTER SERVER dotyczą wyłącznie źródeł danych w systemie stowarzyszonym. Nie można za ich pomocą odwołać się do serwera stowarzyszonego ani do serwerów aplikacji DRDA.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika wydającego instrukcję ALTER SERVER musi mieć przyznane uprawnienie SYSADM lub DBADM względem stowarzyszonej bazy danych.

Ograniczenia:

W instrukcji ALTER SERVER nie można określić opakowania, które nie zostało zarejestrowane na serwerze stowarzyszonym.

Serwer stowarzyszony nie może przetworzyć instrukcji ALTER SERVER w ramach danej jednostki pracy, gdy spełniony jest jeden z następujących warunków:

- Instrukcja odwołuje się do pojedynczego źródła danych, a dana jednostka pracy zawiera już jedną z poniższych instrukcji:
 - instrukcję SELECT, która odwołuje się do pseudonimu tabeli lub widoku ze źródła danych;
 - otwarty kursor dla pseudonimu tabeli lub widoku ze źródła danych;
 - instrukcję wstawienia, usunięcia lub aktualizacji dla pseudonimu tabeli lub widoku ze źródła danych.
- Instrukcja odwołuje się do kategorii źródeł danych (na przykład wszystkich źródeł danych określonego typu i wersji), a jednostka pracy zawiera już jedną z następujących instrukcji:
 - instrukcję SELECT, która odwołuje się do pseudonimu tabeli lub widoku w jednym ze źródeł danych;
 - otwarty kursor dla pseudonimu tabeli lub widoku w jednym ze źródeł danych;
 - instrukcję wstawienia, usunięcia lub aktualizacji dla pseudonimu tabeli lub widoku w jednym ze źródeł danych.

Definicję serwera należy zmienić w następujących sytuacjach:

- nastąpiło uaktualnienie do nowej wersji źródła danych;
- należy wprowadzić identyczną zmianę we wszystkich definicjach serwerów z określonym typem źródła danych;
- należy dodać lub zmienić opcję serwera w istniejącej definicji serwera.

Zadania pokrewne:

- “Modyfikowanie wersji źródła danych w definicji serwera” na stronie 29
- “Usuwanie definicji serwera” na stronie 42

Informacje pokrewne:

- Rozdział 21, “Opcje serwera w systemie stowarzyszonym”, na stronie 225

Modyfikowanie definicji serwera i opcji serwera - informacje szczegółowe

Do modyfikowania definicji serwera służy instrukcja ALTER SERVER.

Modyfikowanie wersji źródła danych w definicji serwera

Istniejącą definicję serwera można zmodyfikować, zmieniając zapisaną w tej definicji wersję źródła danych udostępnianego przez zdalny serwer.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika wydającego instrukcję ALTER SERVER musi mieć przyznane uprawnienie SYSADM lub DBADM względem stowarzyszonej bazy danych.

Procedura:

Definicję serwera można zmodyfikować przy użyciu Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Rozwiń folder Obiekty stowarzyszonej bazy danych. Obiekty definicji serwera zostaną wyświetlone w panelu zawartości, w oknie Centrum sterowania DB2.
2. Kliknij prawym przyciskiem myszy definicję serwera, którą chcesz zmienić, a następnie pozycję **Zmień** na liście działań. Zostanie otwarty notatnik Modyfikuj definicję serwera.
3. Na stronie Serwer kliknij strzałkę **Wersja**, aby określić inną wersję źródła danych.
4. Kliknij przycisk **OK**, aby zmodyfikować definicję serwera i zamknąć notatnik Modyfikuj definicję serwera.

Aby wykonać to zadanie z wiersza komend DB2:

Użyj instrukcji ALTER SERVER.

Załóżmy, że w systemie stowarzyszonym zdefiniowane jest źródło danych Microsoft SQL Server wersja 6.5. W instrukcji CREATE SERVER serwerowi przypisano nazwę SQLSVR_ASIA. Po uaktualnieniu serwera Microsoft SQL Server do wersji 7.0 należy zmodyfikować odpowiednio definicję serwera, wydając następującą instrukcję:

```
ALTER SERVER SQLSVR_ASIA VERSION 7
```

Zadania pokrewne:

- “Modyfikowanie wszystkich definicji serwerów dla określonego typu źródła danych” na stronie 30

Informacje pokrewne:

- “ALTER SERVER statement” w podręczniku *SQL Reference, Volume 2*

Modyfikowanie wszystkich definicji serwerów dla określonego typu źródła danych

Za pomocą jednej instrukcji ALTER SERVER można zmodyfikować wszystkie istniejące definicje serwerów dla określonego typu źródła danych. Jest to wygodne w sytuacjach, gdy trzeba wprowadzić identyczną zmianę we wszystkich definicjach serwerów danego typu.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika wydającego instrukcję ALTER SERVER musi mieć przyznane uprawnienie SYSADM lub DBADM względem stowarzyszonej bazy danych.

Ograniczenia:

Jeśli opcje serwera zostały wcześniej dodane przy użyciu instrukcji ALTER SERVER, można je ustawiać lub usuwać wyłącznie przy użyciu instrukcji ALTER SERVER dla wszystkich źródeł danych określonego typu.

Procedura:

Załóżmy, że w katalogu globalnym zarejestrowanych jest pięć serwerów Sybase dla źródeł danych Sybase. Naszym celem jest uzyskanie takiej konfiguracji, w której identyfikatory wysyłane przez serwer stowarzyszony do wszystkich tych serwerów Sybase w celu

uwierzytelnienia zawsze będą przekształcane na wielkie litery. Ponadto należy określić, jak długo serwer stowarzyszony może oczekiwać na odpowiedź tych serwerów Sybase na instrukcje SQL. Czas określany jest w sekundach.

Poniższa instrukcja pozwala zmodyfikować wszystkie pięć definicji serwerów jednocześnie:

```
ALTER SERVER TYPE sybase
  OPTIONS (ADD FOLD_ID 'U', ADD TIMEOUT '600')
```

Zadania pokrewne:

- “Modyfikowanie wersji źródła danych w definicji serwera” na stronie 29

Informacje pokrewne:

- “ALTER SERVER statement” w podręczniku *SQL Reference, Volume 2*

Korzystanie z opcji serwera w definicjach serwera

Opcje serwera dzielą się na opcje o zastosowaniu ogólnym oraz opcje dotyczące tylko określonych typów źródeł danych. Definicje serwerów mogą być modyfikowane w drodze dodawania lub modyfikowania opcji serwera.

Opcjom serwera nadaje się wartości, które są przechowywane między kolejnymi połączeniami z serwerem. Wartości te są przechowywane w katalogu systemu stowarzyszonego.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika wydającego instrukcję ALTER SERVER musi mieć przyznane uprawnienie SYSADM lub DBADM względem stowarzyszonej bazy danych.

Procedura:

Definicję serwera można zmodyfikować przy użyciu Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Rozwiń folder Obiekty stowarzyszonej bazy danych. Obiekty definicji serwera zostaną wyświetlone w panelu zawartości, w oknie Centrum sterowania DB2.
2. Kliknij prawym przyciskiem myszy definicję serwera, którą chcesz zmienić, a następnie pozycję **Zmień** na liście działań. Zostanie otwarty notatnik Modyfikuj definicję serwera.
3. Na stronie Ustawienia zaznacz opcję serwera, którą chcesz dodać lub usunąć.
4. Dla opcji dodawanych lub zmienianych podaj wartość opcji.
5. Kliknij przycisk **OK**, aby zmodyfikować definicję serwera i zamknąć notatnik Modyfikuj definicję serwera.

Niektóre opcje serwera są wymagane i nie można ich usunąć. Z kolei innych opcji serwera nie można dodawać, gdy ustawione są już konkretne opcje. Listę opisów wszystkich opcji można znaleźć w sekcji Opcje serwera dla systemów stowarzyszonych.

Aby wykonać to zadanie z wiersza komend DB2:

Użyj instrukcji ALTER SERVER. Poniżej przedstawiono kilka przykładów opcji serwera:

- Załóżmy, że utworzono definicję serwera Informix, nadając mu nazwę INFMX01. Teraz trzeba nadać opcji DB2_MAXIMAL_PUSHDOWN wartość Y. Instrukcja modyfikująca definicję serwera wygląda następująco:

```
ALTER SERVER INFMX01 OPTIONS (SET DB2_MAXIMAL_PUSHDOWN 'Y')
```

- Załóżmy, że utworzono definicję serwera Oracle, nadając mu nazwę ORCL99. Teraz zachodzi potrzeba dodania do definicji opcji FOLD_ID i FOLD_PW. Instrukcja zmieniająca definicję serwera miałaby następującą postać:

```
ALTER SERVER ORCL99 OPTIONS (ADD FOLD_ID 'U', FOLD_PW 'U')
```
- Załóżmy, że ustawienie limitu czasu powinno mieć wartość równą liczbie sekund, przez jaką opakowanie CTLIB powinno oczekiwać na odpowiedź z serwera Sybase. Do ustawiania tej wartości służy opcja serwera TIMEOUT. Instrukcja zmieniająca definicję serwera miałaby następującą postać:

```
ALTER SERVER SYBSERVER OPTIONS (ADD TIMEOUT '60')
```

Tymczasowa zmiana opcji serwera dla relacyjnych źródeł danych

Aby tymczasowo ustawić opcje serwera dla relacyjnego źródła danych, należy użyć instrukcji SET SERVER OPTION. Wartość ustawiona w ten sposób zastępuje opcję serwera określoną w definicji serwera na czas pojedynczego połączenia ze stowarzyszoną bazą danych. Wartość zastępująca nie jest wprowadzana do katalogu globalnego.

Oto przykładowa instrukcja SET SERVER OPTION:

```
SET SERVER OPTION PLAN_HINTS TO 'Y' FOR SERVER ORA_SERVER
```

Instrukcja SET SERVER OPTION używana ze statycznymi instrukcjami SQL nie będzie miała żadnego wpływu na opcję serwera IUD_APP_SVPT_ENFORCE.

Hierarchia ustawień opcji serwera

Gdy jedna i ta sama opcja serwera ma określone ustawienie dla pewnego typu źródła danych oraz inne ustawienie dla określonego serwera źródła danych, ustawienia te mają określoną hierarchię. Załóżmy na przykład, że opcji serwera PLAN_HINTS nadano ustawienie 'Y' dla typu źródła danych ORACLE. Jednak ta sama opcja, PLAN_HINTS, ma ustawienie 'N' w definicji konkretnego serwera Oracle o nazwie PURNELL. Ustawienie dotyczące konkretnego serwera źródła danych ma wyższy priorytet niż ogólne ustawienie dla typu źródła danych. W opisanej konfiguracji opcja PLAN_HINTS jest włączona na wszystkich serwerach źródeł danych Oracle z wyjątkiem serwera PURNELL.

Pojęcia pokrewne:

- “Definicje serwerów i opcje serwera” na stronie 13

Informacje pokrewne:

- Rozdział 21, “Opcje serwera w systemie stowarzyszonym”, na stronie 225
- “ALTER SERVER statement” w podręczniku *SQL Reference, Volume 2*
- “SET SERVER OPTION statement” w podręczniku *SQL Reference, Volume 2*

Modyfikowanie odwzorowania użytkowników

Odwzorowanie użytkowników jest to skojarzenie między identyfikatorem autoryzowanego użytkownika na serwerze stowarzyszonym a identyfikatorem autoryzowanego użytkownika w źródle danych. Odwzorowania użytkowników są niezbędne do wysyłania do źródeł danych żądań rozproszonych.

Instrukcja ALTER USER MAPPING służy do zmiany identyfikatorów autoryzowanego użytkownika lub haseł, które w źródle danych odpowiadają określonemu identyfikatorowi autoryzowanego użytkownika na serwerze stowarzyszonym.

Wymagania wstępne:

Jeśli identyfikator autoryzowanego użytkownika wydającego instrukcję jest różny od identyfikatora odwzorowanego w źródle danych, identyfikator autoryzowanego użytkownika wydającego instrukcję musi mieć uprawnienie SYSADM lub DBADM w stowarzyszonej bazie danych.

Ograniczenia:

Serwer stowarzyszony nie może przetworzyć instrukcji ALTER USER w ramach danej jednostki pracy, gdy jednostka ta zawiera już jedną z następujących instrukcji:

- instrukcję SELECT, która odwołuje się do pseudonimu tabeli lub widoku ze źródła danych objętego odwzorowaniem;
- otwarty kursor dla pseudonimu tabeli lub widoku ze źródła danych objętego odwzorowaniem;
- instrukcję wstawienia, usunięcia lub aktualizacji dla pseudonimu tabeli lub widoku źródła danych objętego odwzorowaniem.
-

Procedura:

Odwzorowanie użytkownika można zmodyfikować przy użyciu Centrum sterowania DB2 lub z wiersza komend.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Rozwiń folder Obiekty stowarzyszonej bazy danych. Obiekty odwzorowań użytkowników zostaną wyświetlone w panelu zawartości, w oknie Centrum sterowania DB2.
2. Kliknij prawym przyciskiem myszy odwzorowanie użytkownika, które chcesz zmienić, a następnie pozycję **Zmień** na liście działań. Zostanie otwarty notatnik Zmień odwzorowanie użytkownika.
3. Zmień wartość opcji.
4. Kliknij przycisk **OK**, aby zmodyfikować odwzorowanie i zamknąć notatnik Zmień odwzorowanie użytkownika.

Aby wykonać to zadanie z wiersza komend DB2:

Użyj instrukcji ALTER USER MAPPING.

Na przykład Jenny korzysta z serwera stowarzyszonego, by łączyć się z serwerem Sybase o nazwie SYBSERVER. Dostęp do serwera stowarzyszonego uzyskuje za pomocą identyfikatora autoryzowanego użytkownika *jennifer*. Identyfikator *jennifer* jest odwzorowany na identyfikator *jenn* na serwerze Sybase. Identyfikator autoryzowanego użytkownika dla Jenny na serwerze Sybase zostanie zmieniony na *jen123*. Instrukcja ALTER USER MAPPING odwzorowująca identyfikator *jennifer* na *jen123* ma postać:

```
ALTER USER MAPPING FOR jennifer SERVER SYBSERVER  
OPTIONS (SET REMOTE_AUTHID 'jen123')
```

Tomas korzysta z serwera stowarzyszonego, aby łączyć się z serwerem Oracle o nazwie ORASERVER. Dostęp do serwera stowarzyszonego uzyskuje za pomocą identyfikatora autoryzowanego użytkownika *tomas*. Identyfikator autoryzowanego użytkownika *tomas* jest odwzorowany na identyfikator autoryzowanego użytkownika *tom* na serwerze Oracle. Hasło Tomasa na serwerze Oracle zostało zmienione. Jego nowe hasło brzmi *day2night*. Instrukcja ALTER USER MAPPING odwzorowująca identyfikator *tomas* na nowe hasło ma postać:

```
ALTER USER MAPPING FOR tomas SERVER ORASERVER
OPTIONS (SET REMOTE_PASSWORD 'day2night')
```

W ustawieniach opcji użytkowników REMOTE_AUTHID i REMOTE_PASSWORD są rozróżniana jest wielkość liter, chyba że opcje serwera FOLD_ID i FOLD_PW zostaną ustawione na 'U' lub 'L' za pomocą instrukcji CREATE SERVER.

Informacje pokrewne:

- “ALTER USER MAPPING statement” w podręczniku *SQL Reference, Volume 2*
- Rozdział 22, “Opcje odwzorowań użytkowników w systemach stowarzyszonych”, na stronie 241

Modyfikowanie pseudonimu

Pseudonimy to identyfikatory używane w odwołaniach do obiektów źródła danych.

Modyfikacja pseudonimu może być potrzebna w celu:

- zmiany lokalnych nazw kolumn obiektu źródła danych,
- zmiany lokalnych typów danych kolumn obiektu źródła danych,
- dodania, ustawienia lub usunięcia opcji pseudonimu i opcji kolumn,
- dodania lub usunięcia klucza podstawowego,
- dodania lub usunięcia ograniczeń kluczy, ograniczeń referencyjnych lub ograniczeń sprawdzających,
- zmiany atrybutów ograniczeń referencyjnych, sprawdzających lub zależności funkcjonalnej.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika skojarzony z instrukcją musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie ALTER względem pseudonimu występującego w instrukcji
- uprawnienie CONTROL względem pseudonimu występującego w instrukcji
- uprawnienie ALTERIN względem schematu, o ile nazwa schematu pseudonimu istnieje
- uprawnienia użytkownika definiującego pseudonim, zgodnie z zawartością kolumny DEFINER w widoku katalogu pseudonimu.

Ograniczenia:

Zobacz temat poświęcony ograniczenia dotyczące definiowania pseudonimów.

Procedura:

Pseudonim można zmodyfikować przy użyciu Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Wybierz folder **Pseudonimy**.
2. Kliknij prawym przyciskiem myszy pseudonim, który chcesz zmienić, i kliknij przycisk **Zmień**. Zostanie otwarty notatnik Zmodyfikuj pseudonim.
3. Na stronie Pseudonimy zmień lokalne nazwy kolumn, lokalne typy danych lub opcje kolumn przechowywane w katalogu globalnym.

4. Na stronie Klucze skonfiguruj ograniczenia spójności referencyjnej dla pseudonimu. Możesz także określić ograniczenie klucza podstawowego, klucza unikalnego lub klucza obcego.
5. Na stronie Ograniczenia sprawdzające określ ograniczenia sprawdzające lub ograniczenia zależności funkcjonalnej dla pseudonimu.
6. Na stronie Ustawienia określ opcje pseudonimu.
7. Kliknij przycisk **OK**, aby zmodyfikować pseudonim i zamknąć notatnik.
Niektóre opcje pseudonimu są wymagane i nie można ich usunąć. Z kolei innych opcji pseudonimu nie można dodawać, gdy ustawione są już konkretne opcje. Listę opisów wszystkich opcji można znaleźć w sekcjach Opcje pseudonimów dla systemów stowarzyszonych i Opcje kolumn pseudonimów dla systemów stowarzyszonych.

Aby wykonać to zadanie z wiersza komend DB2, należy użyć instrukcji ALTER NICKNAME z odpowiednimi parametrami.

Gdy struktura lub zawartość obiektu źródła danych znacznie się zmieni, należy zaktualizować statystyki pseudonimu. Znaczne zmiany to na przykład dodanie lub usunięcie wielu wierszy.

Pojęcia pokrewne:

- “Ograniczenia informacyjne pseudonimów” na stronie 185
- “Narzędzie do aktualizacji statystyk pseudonimu - przegląd” na stronie 191

Zadania pokrewne:

- “Modyfikowanie opcji pseudonimów” na stronie 38
- “Zmiana typu lokalnego dla obiektu źródła danych” na stronie 54
- “Modyfikowanie nazw kolumn pseudonimów” na stronie 37
- “Modyfikowanie opcji kolumn pseudonimów” na stronie 39

Informacje pokrewne:

- “Ograniczenia dotyczące zmiany pseudonimów” na stronie 35
- Rozdział 23, “Opcje pseudonimu dla systemów stowarzyszonych”, na stronie 243
- Rozdział 24, “Opcje kolumn pseudonimu dla systemów stowarzyszonych”, na stronie 253
- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*

Modyfikowanie pseudonimów - informacje szczegółowe

Użytkownik może zmieniać nazwy kolumn źródeł danych przechowywane w globalnym katalogu i ustawiać opcje kolumn, zmieniając ich pseudonimy.

Ograniczenia dotyczące zmiany pseudonimów

Zmiany pseudonimów podlegają następującym ograniczeniom:

Nazwy kolumn

Nie można użyć instrukcji ALTER NICKNAME do zmiany nazw kolumn wymienionych poniżej źródeł danych. Konieczne jest usunięcie pseudonimu i jego ponowne utworzenie z poprawnymi nazwami kolumn.

- BLAST
- Documentum
- HMMER

Opcje kolumny

Jeśli dla danej kolumny ustawiono jedną z poniższych opcji, nie można dodać innych opcji dla tej kolumny:

- SOAPACTIONCOLUMN
- URLCOLUMN
- PRIMARY_KEY
- FOREIGN_KEY

Dotyczy źródeł danych BioRS

- W razie zmiany nazwy elementu kolumny przy użyciu opcji ELEMENT_NAME, nie jest sprawdzana poprawność nowej nazwy. Niepoprawna opcja może spowodować błędy, gdy w zapytaniu wystąpi odwołanie do tej kolumny.
- Zmiany wprowadzone w opcji kolumny IS_INDEXED nie są weryfikowane przez serwer BioRS. Niepoprawna opcja może spowodować błędy, gdy w zapytaniu wystąpi odwołanie do tej kolumny.

Typy danych

- W razie zmiany typu danych kolumny, nowy typ danych musi być zgodny z typem danych odpowiedniej kolumny lub elementu źródła danych. Zmiana lokalnego typu danych na typ niezgodny ze zdalnym typem danych może spowodować nieprzewidywalne błędy.
- *Lokalny_typ_danych* nie może być typem long varchar, long vargraphic, DATALINK ani typem danych zdefiniowanym przez użytkownika.
- *Typ_danych_źródła_danych* nie może być typem zdefiniowanym przez użytkownika.
- Dla niektórych nierelacyjnych źródeł danych nie można przesłaniać ani tworzyć istniejących typów lokalnych. Więcej informacji na temat tych ograniczeń można znaleźć w dokumentacji konkretnego opakowania źródła danych.
- Po zmianie lokalnej specyfikacji typu danych kolumny menedżer stowarzyszonej bazy danych unieważnia wszelkie statystyki (na przykład HIGH2KEY i LOW2KEY) zgromadzone dla tej kolumny.
- Lokalny typ jest ustawiany dla określonego obiektu źródła danych w momencie uzyskania dostępu do tego obiektu za pośrednictwem pseudonimu. Ten sam obiekt źródła danych może mieć inny pseudonim, korzystający z domyślnego odwzorowania typów danych.

Indeksy

Instrukcja ALTER NICKNAME nie może służyć do rejestrowania nowego indeksu źródła danych w stowarzyszonej bazie danych. Specyfikację indeksu tworzy się za pomocą instrukcji CREATE INDEX z klauzulą SPECIFICATION ONLY.

Parametry LOCAL NAME i LOCAL TYPE

- Nie można użyć instrukcji ALTER NICKNAME do zmiany lokalnych nazw ani typów danych kolumn pseudonimu, gdy:
 - pseudonim jest używany w widoku, metodzie lub funkcji SQL;
 - zdefiniowano ograniczenie informacyjne dla pseudonimu.
- Jeśli w instrukcji ALTER NICKNAME stosowany będzie parametr LOCAL NAME, LOCAL TYPE lub obydwa te parametry, klauzula federated_column_options musi być użyta jako ostatnia.

Pseudonimy

Nie można użyć instrukcji ALTER NICKNAME do zmiany nazwy banku danych BioRS, do którego odwołuje się pseudonim ze źródła danych BioRS lub który jest

używany przez ten pseudonim. Jeśli nazwa banku danych BioRS zmieni się, konieczne będzie usunięcie pseudonimu i jego ponowne utworzenie.

Jednostki pracy

Serwer stowarzyszony nie może przetworzyć instrukcji ALTER NICKNAME w ramach danej jednostki pracy, gdy wystąpi jeden z następujących warunków:

- W tej samej jednostce pracy jest otwarty kursor dla pseudonimu, do którego odwołuje się instrukcja ALTER NICKNAME.
- W tej samej jednostce pracy zostanie użyta instrukcja wstawienia, usunięcia lub aktualizacji dla pseudonimu, do którego występuje odwołanie w instrukcji ALTER NICKNAME.
- W przypadku nierelacyjnych źródeł danych, instrukcja ALTER NICKNAME odwołuje się do pseudonimu, do którego odwołuje się także instrukcja SELECT w tej samej jednostce pracy.

Zadania pokrewne:

- “Modyfikowanie opcji pseudonimów” na stronie 38
- “Zmiana typu lokalnego dla obiektu źródła danych” na stronie 54
- “Modyfikowanie pseudonimu” na stronie 34
- “Modyfikowanie nazw kolumn pseudonimów” na stronie 37
- “Modyfikowanie opcji kolumn pseudonimów” na stronie 39

Modyfikowanie nazw kolumn pseudonimów

Po utworzeniu pseudonimu nazwy kolumn skojarzone z obiektem źródła danych są przechowywane w stowarzyszonej bazie danych. W przypadku niektórych źródeł danych nazwy kolumn są określane przez opakowanie. W przypadku innych źródeł danych należy określić nazwy kolumn samodzielnie podczas tworzenia pseudonimu.

Aby zmienić nazwy kolumn, można zmodyfikować pseudonim.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika uruchamiającego instrukcję musi mieć przyznane co najmniej jedno z następujących uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie ALTER względem pseudonimu występującego w instrukcji
- uprawnienie CONTROL względem pseudonimu występującego w instrukcji
- uprawnienie ALTERIN względem schematu, o ile nazwa schematu pseudonimu istnieje
- uprawnienia użytkownika definiującego pseudonim, zgodnie z zawartością kolumny DEFINER w widoku katalogu pseudonimu.

Ograniczenia:

Zobacz temat poświęcony ograniczenia dotyczące definiowania pseudonimów.

Procedura:

Nazwy kolumn można zmieniać korzystając z Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Wybierz folder **Pseudonimy**.

2. Kliknij prawym przyciskiem myszy pseudonim, który chcesz zmienić, i kliknij przycisk **Zmień**. Zostanie otwarty notatnik Zmodyfikuj pseudonim.
3. Na stronie Pseudonimy zaznacz kolumnę, którą chcesz zmienić, i kliknij przycisk **Zmień**. Zostanie otwarte okno Zmień kolumnę.
4. Wpisz nazwę kolumny.
5. Kliknij przycisk **OK**, aby zmienić nazwę kolumny i zamknąć okno.
6. Kliknij przycisk **OK**, aby zmodyfikować pseudonim i zamknąć notatnik.

Aby wykonać to zadanie z wiersza komend programu DB2, należy użyć instrukcji ALTER NICKNAME:

```
ALTER NICKNAME pseudonim  
ALTER COLUMN bieżąca_nazwa  
LOCAL NAME nowa_nazwa
```

Przykład: Zmiana lokalnej nazwy kolumny pseudonimu:

Niech na przykład pseudonim Z_EMPLOYEES tabeli z bazy danych DB2 UDB for z/OS zawiera kolumnę o nazwie EMPNO. Aby użytkownicy mogli posługiwać się nazwą *Numer_pracownika* zamiast *EMPNO*, można użyć następującej instrukcji:

```
ALTER NICKNAME Z_EMPLOYEES ALTER COLUMN EMPNO  
LOCAL NAME "Numer_pracownika"
```

Zadania pokrewne:

- “Modyfikowanie pseudonimu” na stronie 34

Informacje pokrewne:

- “Ograniczenia dotyczące zmiany pseudonimów” na stronie 35
- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*

Modyfikowanie opcji pseudonimów

Opcje pseudonimów to parametry, które określa się dla pseudonimu w instrukcjach CREATE NICKNAME i ALTER NICKNAME.

Za pomocą instrukcji ALTER NICKNAME można dodawać, ustawiać i usuwać opcje pseudonimów.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika uruchamiającego instrukcję musi mieć przyznane co najmniej jedno z następujących uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie ALTER względem pseudonimu występującego w instrukcji
- uprawnienie CONTROL względem pseudonimu występującego w instrukcji
- uprawnienie ALTERIN względem schematu, o ile nazwa schematu pseudonimu istnieje
- uprawnienia użytkownika definiującego pseudonim, zgodnie z zawartością kolumny DEFINER w widoku katalogu pseudonimu.

Ograniczenia:

Patrz temat poświęcony ograniczeniom dotyczącym modyfikowania pseudonimów.

Procedura:

Nazwy kolumn można zmieniać korzystając z Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Wybierz folder **Pseudonimy**.
2. Kliknij prawym przyciskiem myszy pseudonim, który chcesz zmienić, i kliknij przycisk **Zmień**. Zostanie otwarty notatnik Zmodyfikuj pseudonim.
3. Na stronie Ustawienia zaznacz pole wyboru obok każdej opcji, którą chcesz dodać lub usunąć. Nie można usunąć opcji wymaganej.
4. Aby określić lub zmienić wartość opcji, kliknij pole **Wartość** dla tej opcji. Zależnie od opcji, można wybrać wartość z listy, zaznaczyć kilka wartości lub wpisać nową wartość.
5. Kliknij przycisk **OK**, aby zmodyfikować pseudonim i zamknąć notatnik.

Aby wykonać to zadanie z wiersza komend, należy użyć instrukcji ALTER NICKNAME. Na przykład:

```
ALTER NICKNAME pseudonim  
  OPTIONS (SET nazwa_opcji 'wartość_łańcucha_opcji')
```

Na przykład pseudonim DRUGDATA1 jest tworzony dla pliku drugdata1.txt o strukturze tabeli. Pełna ścieżka zdefiniowana pierwotnie w instrukcji CREATE NICKNAME to /user/pat/drugdata1.txt.

Aby zmienić opcję FILE_PATH pseudonimu, należy użyć następującej instrukcji:

```
ALTER NICKNAME DRUGDATA1 OPTIONS (SET FILE_PATH '/usr/kelly/data/drugdata1.txt')
```

Zadania pokrewne:

- “Modyfikowanie pseudonimu” na stronie 34

Informacje pokrewne:

- “Ograniczenia dotyczące zmiany pseudonimów” na stronie 35
- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*
- Rozdział 23, “Opcje pseudonimu dla systemów stowarzyszonych”, na stronie 243

Modyfikowanie opcji kolumn pseudonimów

Informacje o kolumnach w instrukcjach CREATE NICKNAME i ALTER NICKNAME określa się przy użyciu parametrów zwanych *opcjami kolumn pseudonimu*. Wartości tych opcji mogą być pisane małymi lub wielkimi literami.

Używając instrukcji ALTER NICKNAME można dodawać, ustawiać i usuwać opcje kolumn pseudonimu.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika uruchamiającego instrukcję musi mieć przyznane co najmniej jedno z następujących uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie ALTER względem pseudonimu występującego w instrukcji
- uprawnienie CONTROL względem pseudonimu występującego w instrukcji
- uprawnienie ALTERIN względem schematu, o ile nazwa schematu pseudonimu istnieje

- uprawnienia użytkownika definiującego pseudonim, zgodnie z zawartością kolumny DEFINER w widoku katalogu pseudonimu.

Ograniczenia:

Zobacz temat poświęcony ograniczenia dotyczące definiowania pseudonimów.

Procedura:

Nazwy kolumn można zmieniać korzystając z Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Wybierz folder **Pseudonimy**.
2. Kliknij prawym przyciskiem myszy pseudonim, który chcesz zmienić, i kliknij przycisk **Zmień**. Zostanie otwarty notatnik Zmodyfikuj pseudonim.
3. Na stronie Pseudonimy zaznacz kolumnę, którą chcesz zmienić, i kliknij przycisk **Zmień**. Zostanie otwarte okno Zmień kolumnę.
4. Na stronie Ustawienia zaznacz opcję kolumny, którą chcesz dodać lub usunąć.
5. Dla opcji dodawanych lub zmienianych podaj wartość opcji.
6. Kliknij przycisk **OK**, aby zmienić opcję kolumny i zamknąć okno.
7. Kliknij przycisk **OK**, aby zmodyfikować pseudonim i zamknąć notatnik.

Aby wykonać to zadanie z wiersza komend, należy użyć instrukcji ALTER NICKNAME.

Przykład 1: Określanie opcji kolumny NUMERIC_STRING dla relacyjnych źródeł danych:

Opcja kolumny NUMERIC_STRING ma zastosowanie do kolumn typu znakowego (CHAR i VARCHAR). Załóżmy, że w źródle danych ustawiona jest inna kolejność zestawiania niż w stowarzyszonej bazie danych. Serwer stowarzyszony w takiej sytuacji z reguły nie przekazywałby wykonania operacji sortowania kolumn z danymi znakowymi do źródła danych. Dane byłyby przekazywane do stowarzyszonej bazy danych i sortowane lokalnie. Załóżmy jednak, że kolumna ma przypisany typ znakowy, mimo że zawiera wyłącznie dane złożone z cyfr ('0', '1', ..., '9'). Informację o tym można przekazać, ustawiając opcję kolumny NUMERIC_STRING na 'Y'. Dzięki temu optymalizator zapytań DB2 UDB uzyskuje możliwość realizacji sortowania w źródle danych. Realizując sortowanie zdalnie, można uniknąć narzutu związanego z przekazywaniem danych na serwer stowarzyszony i wykonywaniem sortowania lokalnie.

Pseudonim ORA_INDSALES dotyczy tabeli Oracle o nazwie INDONESIA_SALES. Tabela zawiera kolumnę POSTAL_CODE typu VARCHAR. Pierwotnie kolumna ta zawierała wyłącznie cyfry, toteż opcja NUMERIC_STRING miała dla niej ustawienie 'Y'. Teraz jednak kolumna zawiera oprócz cyfr także inne znaki. Aby zmienić wartość opcji kolumny NUMERIC_STRING na 'N', należy użyć instrukcji:

```
ALTER NICKNAME ORA_INDSALES ALTER COLUMN POSTAL_CODE
  OPTIONS (SET NUMERIC_STRING 'N')
```

Przykład 2: Określanie opcji kolumny VARCHAR_NO_TRAILING_BLANKS dla relacyjnych źródeł danych:

Opcja kolumny VARCHAR_NO_TRAILING_BLANKS służy do identyfikowania konkretnych kolumn, które nie zawierają końcowych znaków odstępu. Kompilator SQL

uwzględni to ustawienie, analizując możliwości wykonywania różnych operacji na kolumnach, takich jak porównywanie łańcuchów.

ORA_INDSALES jest pseudonimem tabeli Oracle o nazwie INDONESIA_SALES. Tabela zawiera kolumnę NAME typu znaków odstępu VARCHAR. Wartości w kolumnie NAME nie zawierają końcowych znaków odstępu. Aby dodać opcję VARCHAR_NO_TRAILING_BLANKS do pseudonimu, należy użyć instrukcji:

```
ALTER NICKNAME ORA_INDSALES ALTER COLUMN NAME
  OPTIONS (ADD VARCHAR_NO_TRAILING_BLANKS 'Y')
```

Przykład 3: Określanie opcji kolumny XPATH dla nierelacyjnych źródeł danych:

Pseudonim EMPLOYEE jest pseudonimem źródła danych XML. Opcja XPATH została określona dla kolumny *fname*. Aby zmienić wartość opcji kolumny XPATH na inną ścieżkę, należy użyć instrukcji:

```
ALTER NICKNAME EMPLOYEE ALTER COLUMN fname
  OPTIONS (SET XPATH './@first')
```

Zadania pokrewne:

- “Modyfikowanie pseudonimu” na stronie 34

Informacje pokrewne:

- “Ograniczenia dotyczące zmiany pseudonimów” na stronie 35
- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*
- Rozdział 24, “Opcje kolumn pseudonimu dla systemów stowarzyszonych”, na stronie 253

Usuwanie opakowania

Istnieje kilka przyczyn, dla których może zachodzić potrzeba usunięcia opakowania.

Czasami dostęp do źródła danych może być realizowany za pośrednictwem więcej niż jednego opakowania. Wybór opakowania zależy wtedy od wersji używanego oprogramowania klienta źródła danych. Wybór ten może być też uwarunkowany systemem operacyjnym serwera stowarzyszonego. Załóżmy, że wymagany jest dostęp do dwóch tabel Oracle i jednego widoku Oracle. Używane jest źródło Oracle wersja 8, a serwer stowarzyszony działa pod kontrolą systemu Windows NT. Pierwotnie utworzono opakowanie SQLNET. Ponieważ program DB2 Information Integrator nie obsługuje już opakowania SQLNET, można je usunąć i utworzyć opakowanie NET8.

Innym powodem usuwania opakowania może być to, że dostęp do źródła danych, z którym opakowanie jest skojarzone, przestaje być konieczny. Dla przykładu załóżmy, że wymagany jest dostęp kliencki do baz danych Informix i Microsoft SQL Server. Utworzono jedno opakowanie dla źródła danych Informix i drugie dla źródła Microsoft SQL Server. Następnie jednak w instytucji podjęto decyzję o przeniesieniu wszystkich danych z bazy Microsoft SQL Server do bazy Informix. Opakowanie Microsoft SQL Server przestaje być tym samym potrzebne i można je usunąć.

Uwaga: Usunięcie opakowania łączy się z poważnymi konsekwencjami. Operacja ta ma wpływ na inne obiekty zarejestrowane na serwerze stowarzyszonym:

- Usuwane są wszystkie definicje serwerów, odwzorowania funkcji zdefiniowanych przez użytkownika oraz typów danych zdefiniowanych przez użytkownika, które są zależne od usuwanego opakowania.

- Usunięcie definicji serwerów zależnych od opakowania wpływa na obiekty zależne od tych definicji serwerów. Usunięte zostają wszystkie odwzorowania funkcji zdefiniowanej przez użytkownika, odwzorowania typów i odwzorowania użytkowników zależne od usuwanych definicji serwerów.
- Usuwane są wszystkie pseudonimy zależne od usuwanych definicji serwerów. Usunięcie pseudonimów zależnych od definicji serwerów ma z kolei wpływ na obiekty zależne od tych pseudonimów:
 - Usuwane są wszystkie specyfikacje indeksów zależne od usuniętych pseudonimów.
 - Wszelkie widoki stowarzyszone zależne od usuniętych pseudonimów są oznaczane jako nieoperatywne.
 - Usunięte zostaną także wszystkie zmaterializowane tabele zapytań zależne od usuniętych pseudonimów.
- Wszystkie aplikacje zależne od usuniętych obiektów i widoków nieoperatywnych zostają unieważnione.

Wymagania wstępne:

Do uruchomienia instrukcji DROP WRAPPER niezbędne jest uprawnienie SYSADM lub DBADM.

Procedura:

Aby usunąć opakowanie, należy użyć instrukcji DROP. Na przykład instrukcja usuwająca opakowanie Microsoft SQL Server *MSSQLODBC3* ma następującą postać:

```
DROP WRAPPER MSSQLODBC3
```

Informacje pokrewne:

- “DROP statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE WRAPPER statement” w podręczniku *SQL Reference, Volume 2*

Usuwanie definicji serwera

Usunięcie definicji serwera polega na usunięciu definicji z katalogu globalnego. Operacja ta nie ma wpływu na obiekt źródła danych, do którego definicja serwera się odwołuje.

Usunięcie definicji serwera ma wpływ na inne obiekty zarejestrowane na serwerze stowarzyszonym:

- Usunięte zostają wszystkie odwzorowania funkcji zdefiniowanej przez użytkownika, odwzorowania typów i odwzorowania użytkowników zależne od usuwanej definicji serwera.
- Usuwane są wszystkie pseudonimy zależne od usuwanej definicji serwera. Usunięcie pseudonimów zależnych od definicji serwera ma z kolei wpływ na obiekty zależne od tych pseudonimów:
 - Usuwane są wszystkie specyfikacje indeksów zależne od usuniętych pseudonimów.
 - Wszelkie widoki stowarzyszone zależne od usuniętych pseudonimów są oznaczane jako nieoperatywne.
- Wszystkie aplikacje zależne od usuniętych obiektów i widoków nieoperatywnych zostają unieważnione.

Do usuwania definicji serwera służy instrukcja DROP.

Wymagania wstępne:

Do usunięcia definicji serwera wymagane jest uprawnienie SYSADM lub DBADM.

Ograniczenia:

Serwer stowarzyszony nie może przetworzyć instrukcji DROP SERVER w ramach danej jednostki pracy, gdy spełniony jest jeden z następujących warunków:

- Instrukcja odwołuje się do pojedynczego źródła danych, a dana jednostka pracy zawiera już jedną z poniższych instrukcji:
 - instrukcję SELECT, która odwołuje się do pseudonimu tabeli lub widoku ze źródła danych;
 - otwarty kursor dla pseudonimu tabeli lub widoku ze źródła danych;
 - instrukcję wstawienia, usunięcia lub aktualizacji dla pseudonimu tabeli lub widoku ze źródła danych.
- Instrukcja odwołuje się do kategorii źródeł danych (na przykład wszystkich źródeł danych określonego typu i wersji), a jednostka pracy zawiera już jedną z następujących instrukcji:
 - instrukcję SELECT, która odwołuje się do pseudonimu tabeli lub widoku w jednym ze źródeł danych;
 - otwarty kursor dla pseudonimu tabeli lub widoku w jednym ze źródeł danych;
 - instrukcję wstawienia, usunięcia lub aktualizacji dla pseudonimu tabeli lub widoku w jednym ze źródeł danych.

Procedura:

Gdy dostęp do serwera źródła danych nie jest już wymagany, definicję serwera można usunąć ze stowarzyszonej bazy danych. W celu usunięcia definicji serwera można posłużyć się Centrum sterowania DB2 lub instrukcją DROP wydaną z procesora wiersza komend DB2.

Instrukcja usuwająca definicję serwera ma postać:

```
DROP SERVER  
nazwa_serwera
```

gdzie *nazwa_serwera* określa usuwaną definicję serwera.

Jeśli uprzednio zdefiniowano serwer Informix o nazwie INFMX01, instrukcja usuwająca go miałaby następującą postać:

```
DROP SERVER INFMX01
```

Zadania pokrewne:

- “Modyfikowanie definicji serwerów i opcji serwera” na stronie 28

Informacje pokrewne:

- “DROP statement” w podręczniku *SQL Reference, Volume 2*

Usuwanie odwzorowania użytkowników

Gdy użytkownik nie potrzebuje już dostępu do zdalnego źródła danych, należy usunąć odwzorowanie użytkowników między serwerem stowarzyszonym a serwerem zdalnego źródła danych.

Jeśli użytkownik jest odwzorowany na więcej niż jednym serwerze źródła danych, wymagane będzie usunięcie każdego odwzorowania osobno.

Wymagania wstępne:

Aby móc wydać instrukcję DROP USER MAPPING za pośrednictwem identyfikatora autoryzowanego użytkownika innego niż podany w odwzorowaniu, identyfikator autoryzowanego użytkownika musi mieć uprawnienie SYSADM lub DBADM. Jeśli identyfikator autoryzowanego użytkownika wydającej instrukcję jest zgodny z identyfikatorem w odwzorowaniu, nie są wymagane żadne uprawnienia.

Procedura:

Aby usunąć odwzorowanie użytkownika, należy użyć instrukcji DROP:

```
DROP USER MAPPING FOR ID_użytkownika SERVER nazwa_lokalna_serwera
```

ID_użytkownika jest identyfikatorem autoryzowanego użytkownika na serwerze stowarzyszonym. *nazwa_lokalna_serwera* jest nazwą lokalną używaną do identyfikowania serwera zdalnego źródła danych w definicji serwera.

Zadania pokrewne:

- “Modyfikowanie odwzorowania użytkowników” na stronie 32

Informacje pokrewne:

- “DROP statement” w podręczniku *SQL Reference, Volume 2*

Usuwanie pseudonimu

Istnieje kilka przyczyn, dla których może zachodzić potrzeba usunięcia pseudonimu. Na przykład:

- W przypadku zasadniczej zmiany zawartości lub struktury obiektu źródła danych, do którego odwołuje się pseudonim, może być uzasadnione usunięcie pseudonimu. Następnie pseudonim można odtworzyć, aby metadane obiektu zostały na nowo wprowadzone do katalogu globalnego.
- Gdy zachodzi potrzeba zmiany nazwy pseudonimu, należy pseudonim usunąć i odtworzyć go pod nową nazwą.
- Wreszcie, pseudonim można usunąć, gdy przestaje być wymagany dostęp do skojarzonego z nim obiektu źródła danych.

Usunięcie pseudonimu oznacza usunięcie informacji o nim z globalnego katalogu na serwerze stowarzyszonym. Operacja ta nie ma wpływu na obiekt źródła danych, do którego pseudonim się odwołuje.

Usunięcie pseudonimu ma wpływ na inne obiekty zarejestrowane na serwerze stowarzyszonym:

- Usunięcie pseudonimu wpływa na obiekty zależne od tego pseudonimu:
 - Usuwane są wszystkie specyfikacje indeksów zależne od usuniętych pseudonimów.
 - Wszelkie widoki stowarzyszone zależne od usuniętych pseudonimów są oznaczane jako nieoperatywne.
- Wszystkie aplikacje zależne od usuniętych obiektów i widoków nieoperatywnych zostają unieważnione.

W celu usunięcia pseudonimu należy posłużyć się instrukcją DROP.

Wymagania wstępne:

Pseudonim musi figurować w katalogu.

Identyfikator autoryzowanego użytkownika skojarzony z instrukcją DROP usuwającą pseudonimy musi mieć przyznane jedno z następujących uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie DROPIN względem schematu pseudonimu;
- uprawnienia użytkownika definiującego pseudonim, zgodnie z zawartością kolumny DEFINER w widoku katalogu pseudonimu.
- uprawnienie CONTROL względem pseudonimu.

Ograniczenia:

W przypadku pseudonimów relacyjnych źródeł danych serwer stowarzyszony nie może przetworzyć instrukcji DROP NICKNAME w ramach danej jednostki pracy, gdy spełniony jest jeden z następujących warunków:

- W tej samej jednostce pracy otwarty jest kursor dla pseudonimu, do którego odwołuje się instrukcja.
- W tej samej jednostce pracy zostanie użyta instrukcja wstawienia, usunięcia lub aktualizacji dla pseudonimu, do którego odwołuje się instrukcja.

W przypadku pseudonimów nierelacyjnych źródeł danych serwer stowarzyszony nie może przetworzyć instrukcji DROP NICKNAME w ramach danej jednostki pracy, gdy spełniony jest jeden z następujących warunków:

- W tej samej jednostce pracy otwarty jest kursor dla pseudonimu, do którego odwołuje się ta instrukcja.
- W ramach tej samej jednostki pracy do pseudonimu, do którego odwołuje się ta instrukcja, odwołuje się także instrukcja SELECT.
- W tej samej jednostce pracy zostanie użyta instrukcja wstawienia, usunięcia lub aktualizacji dla pseudonimu, do którego odwołuje się ta instrukcja.

Procedura:

Aby usunąć pseudonim, należy użyć instrukcji w postaci:

```
DROP  
NICKNAME pseudonim
```

gdzie *pseudonim* określa pseudonim przeznaczony do usunięcia.

Zadania pokrewne:

- “Modyfikowanie pseudonimu” na stronie 34

Informacje pokrewne:

- “DROP statement” w podręczniku *SQL Reference, Volume 2*

Rozdział 3. Odwzorowania typów danych

Opakowania dołączone do programu DB2 Information Integrator zawierają domyślne odwzorowania typów danych źródeł danych na typy danych programu DB2 Universal Database dla systemów Linux, UNIX i Windows.

Ten rozdział zawiera:

- “Odwzorowania typów danych w systemie stowarzyszonym”
- “Odwzorowania typów danych i katalog globalny stowarzyszonej bazy danych” na stronie 48
- “Kiedy należy tworzyć alternatywne odwzorowania typów danych” na stronie 49
- “Odwzorowania typów danych dla nierelacyjnych źródeł danych” na stronie 50
- “Proste i zwrotne odwzorowania typów danych” na stronie 50
- “Tworzenie odwzorowań typów danych” na stronie 51
- “Tworzenie odwzorowania dla typu źródła danych – przykład” na stronie 51
- “Tworzenie odwzorowania dla typu danych i wersji źródła danych – przykład” na stronie 52
- “Tworzenie odwzorowania typów dla wszystkich obiektów źródła danych na serwerze – przykład” na stronie 53
- “Zmiana typu lokalnego dla obiektu źródła danych” na stronie 54
- “Zmiana typu lokalnego dla obiektu źródła danych – przykłady” na stronie 55
- “Zmiany długich typów danych na typy varchar” na stronie 57

Odwzorowania typów danych w systemie stowarzyszonym

Typy danych w źródle danych muszą być odwzorowane na odpowiednie typy danych w bazie DB2®. To odwzorowanie umożliwia serwerowi stowarzyszonemu pobieranie danych ze źródeł danych.

Program DB2 Information Integrator zawiera zestaw domyślnych odwzorowań typów danych dla niektórych źródeł danych. Dla innych źródeł należy samodzielnie zdefiniować odpowiednie odwzorowania. W przypadku nierelacyjnych źródeł danych nie jest możliwe zmodyfikowanie odwzorowań istniejących ani utworzenie nowych.

Oto niektóre przykłady domyślnych odwzorowań typów danych:

- typ Oracle FLOAT jest domyślnie odwzorowany na typ DB2 DOUBLE,
- typ Oracle DATE jest domyślnie odwzorowany na typ DB2 TIMESTAMP,
- typ DATE z programu DB2 Universal Database™ for z/OS and OS/390® jest domyślnie odwzorowywany na typ DATE programu DB2.

Z nowego odwzorowania typów danych będą korzystać jedynie pseudonimy utworzone po zmianie odwzorowania. Pseudonimy utworzone przed zmianą odwzorowania będą korzystały z odwzorowań domyślnych.

Jeśli pseudonimy zostały utworzone wcześniej, istnieją dwa sposoby ich aktualizacji:

- modyfikacja każdego pseudonimu,
- usunięcie i ponowne utworzenie każdego z pseudonimów.

Serwery stowarzyszone DB2 nie obsługują odwzorowań dla następujących typów danych:

- Lokalny typ danych nie może być jednym z następujących typów: LONG VARCHAR, LONG VARGRAPHIC, DATALINK ani typem danych zdefiniowanym przez użytkownika.
- Zdalnym typem danych nie może być typ zdefiniowany przez użytkownika.

Jednak w widoku w zdalnym źródle danych, które jest identyczne ze źródłem danych, można użyć funkcji rzutowania do przekształcenia typu zdefiniowanego przez użytkownika na wbudowany lub systemowy typ danych. Następnie dla tego widoku można utworzyć pseudonim. Jednak takie widoki nie mają statystyk ani indeksów i nie można ich aktualizować.

Pojęcia pokrewne:

- “Odwzorowania typów danych i katalog globalny stowarzyszonej bazy danych” na stronie 48
- “Kiedy należy tworzyć alternatywne odwzorowania typów danych” na stronie 49
- “Odwzorowania typów danych dla nierelacyjnych źródeł danych” na stronie 50
- “Proste i zwrotne odwzorowania typów danych” na stronie 50

Zadania pokrewne:

- “Tworzenie odwzorowań typów danych” na stronie 51

Informacje pokrewne:

- Rozdział 27, “Domyślne proste odwzorowania typów”, na stronie 269
- Rozdział 28, “Domyślne zwrotne odwzorowania typów”, na stronie 285
- “Zmiany długich typów danych na typy varchar” na stronie 57

Odwzorowania typów danych i katalog globalny stowarzyszonej bazy danych

Pisząc instrukcję CREATE NICKNAME, należy określić obiekt w źródle danych reprezentowany przez pseudonim. W większości przypadków serwer stowarzyszony definiuje dla każdej kolumny lub pola w tym obiekcie źródła danych typ danych obsługiwany przez program DB2[®]. W przypadku niektórych nierelacyjnych źródeł danych wymagane jest podanie informacji o typie danych DB2. Takie lokalne definicje typów danych są przechowywane w widoku SYSCAT.COLUMNS globalnego katalogu stowarzyszonej bazy danych.

W przypadku relacyjnych źródeł danych serwer stowarzyszony wyszukuje informacje na temat prostych odwzorowań typów danych w opakowaniach oraz w widoku katalogu SYSCAT.TYPEMAPPINGS, aby ustalić, który z lokalnych typów danych ma być zapisany w widoku katalogu SYSCAT.COLUMNS. Odwzorowania w widoku katalogu SYSCAT.TYPEMAPPINGS mają wyższy priorytet niż odwzorowania domyślne w opakowaniach. Jeśli w celu przesłonięcia odwzorowań domyślnych zostaną utworzone alternatywne odwzorowania typów danych, serwer stowarzyszony będzie używał odwzorowań alternatywnych. Jeśli dla danej kolumny można zastosować kilka odwzorowań, serwer stowarzyszony wybierze najnowsze z nich (utworzone jako ostatnie).

W przypadku nierelacyjnych źródeł danych serwer stowarzyszony wyszukuje informacje na temat odwzorowań typów danych w opakowaniach, aby ustalić, który z lokalnych typów danych ma być zapisany w widoku katalogu SYSCAT.COLUMNS. Stopień, w jakim można modyfikować typy danych definiowane przez opakowanie zmienia się, zależnie od nierelacyjnego źródła danych. Dla niektórych nierelacyjnych źródeł danych nie określa się

żadnych kolumn. Typy danych są definiowane przez opakowanie. Dla innych źródeł danych można przesłonić typy danych. A dla jeszcze innych źródeł danych konieczne jest określenie typów danych w instrukcji CREATE NICKNAME.

Podczas pisania instrukcji przezroczystego kodu DDL CREATE TABLE dla relacyjnych źródeł danych należy w niej określić typy danych programu DB2. Serwer stowarzyszony poszuka informacji o zwrotnych odwzorowaniach typów danych między programem DB2 UDB i źródłem danych. Serwer stowarzyszony będzie szukał tych informacji w opakowaniu i w widoku katalogu SYSCAT.TYPEMAPPINGS.

Wartości zwracane z kolumny źródła danych do stowarzyszonej bazy danych są w pełni zgodne z typem danych DB2, na który kolumna źródła danych jest odwzorowana. Jeśli jest to odwzorowanie domyślne, wartości są także w pełni zgodne z typem źródła danych w odwzorowaniu. Jeśli na przykład w stowarzyszonej bazie danych zostanie zdefiniowana tabela Oracle z kolumną FLOAT, dla tej kolumny automatycznie zostanie zastosowane odwzorowanie domyślne typu FLOAT programu Oracle na typ DOUBLE programu DB2. W rezultacie wartości zwracane z tej kolumny będą w pełni zgodne ze specyfikacjami typów danych FLOAT i DOUBLE.

Pojęcia pokrewne:

- “Odwzorowania typów danych w systemie stowarzyszonym” na stronie 47

Kiedy należy tworzyć alternatywne odwzorowania typów danych

Dla relacyjnych źródeł danych można utworzyć alternatywne odwzorowania typów danych.

Może to być przydatne w następujących sytuacjach:

- W celu przesłonięcia domyślnego odwzorowania typów danych.
W przypadku niektórych opakowań celowa może być zmiana formatu lub długość zwracanych wartości. Można to zrobić, zmieniając typ danych DB2®, z którym wartości muszą być zgodne. Na przykład typ danych Oracle DATE jest używany jako datownik i zawiera informacje o wieku, roku, miesiącu, dniu, godzinie, minucie i sekundzie. Domyślnie typ danych Oracle DATE jest odwzorowany na typ DB2 TIMESTAMP. Aby zwrócić tylko informacje o godzinie, minutach i sekundach, można przesłonić domyślne odwzorowanie typu danych w taki sposób, aby typ danych Oracle DATE był odwzorowywany na typ danych DB2 TIME. Podczas wykonywania zapytań na kolumnach typu DATE w programie Oracle do serwera stowarzyszonego zwracana będzie tylko godzinowa część wartości datowników Oracle.
- Gdy odwzorowanie domyślne nie istnieje
Jeśli dla określonego typu w źródle danych nie jest dostępne odwzorowanie domyślne, konieczne jest samodzielne utworzenie nowego odwzorowania dla takiego typu danych.

Do definiowania nowych odwzorowań typów danych służy instrukcja CREATE TYPE MAPPING. Utworzone odwzorowania są przechowywane w widoku katalogu SYSCAT.TYPEMAPPINGS w stowarzyszonej bazie danych.

Pojęcia pokrewne:

- “Odwzorowania typów danych w systemie stowarzyszonym” na stronie 47

Zadania pokrewne:

- “Tworzenie odwzorowań typów danych” na stronie 51

Odwzorowania typów danych dla nierelacyjnych źródeł danych

W przypadku niektórych nierelacyjnych źródeł danych opakowania nie zawierają definicji odwzorowań typów danych. W przypadku niektórych nierelacyjnych źródeł danych wymagane jest podanie w instrukcji CREATE NICKNAME informacji o lokalnym typie danych.

Poniższe przykłady pokazują, w jaki sposób określa się typy danych kolumn w instrukcji CREATE NICKNAME dla niektórych nierelacyjnych źródeł danych:

```
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')
```

Pojęcia pokrewne:

- “Odwzorowania typów danych w systemie stowarzyszonym” na stronie 47

Zadania pokrewne:

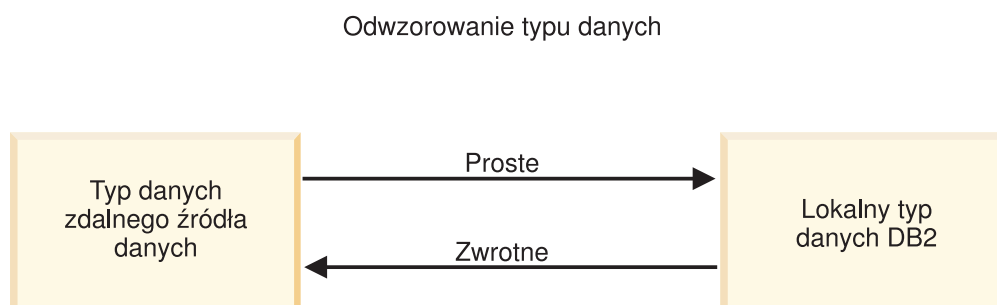
- “Registering nicknames for a data source” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Proste i zwrotne odwzorowania typów danych

W prostym odwzorowaniu typów danych zdalny typ danych jest odwzorowany na porównywalny lokalny typ danych. Odwzorowania proste są używane przy tworzeniu pseudonimu dla obiektu źródła danych. Informacja o odpowiednim lokalnym typie danych dla każdej kolumny w obiekcie źródła danych jest zapisywana w katalogu globalnym.

W zwrotnym odwzorowaniu typów danych lokalny typ danych jest odwzorowany na porównywalny zdalny typ danych. Zwrotne odwzorowanie typów jest stosowane w przezroczystym języku DDL.

Rys. 2 przedstawia proste i zwrotne odwzorowanie typu danych.



Rysunek 2. Proste i zwrotne odwzorowania typów danych

Pojęcia pokrewne:

- “Odwzorowania typów danych w systemie stowarzyszonym” na stronie 47

Informacje pokrewne:

- Rozdział 27, “Domyślne proste odwzorowania typów”, na stronie 269
- Rozdział 28, “Domyślne zwrotne odwzorowania typów”, na stronie 285

Tworzenie odwzorowań typów danych

Do tworzenia nowych odwzorowań typów danych służy instrukcja CREATE TYPE MAPPING. Można ją uruchomić z Centrum komend DB2[®] lub procesora wiersza komend albo dołączyć ją do aplikacji. Nie można tworzyć ani modyfikować odwzorowań typów danych w Centrum sterowania DB2.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Ograniczenia:

- Typ *lokalny_typ_danych* nie może być typem long varchar, long vargraphic, DATALINK ani typem danych zdefiniowanym przez użytkownika.
- Typ *typ_danych_źródła_danych* nie może być typem zdefiniowanym przez użytkownika.
- W przypadku nierelacyjnych źródeł danych nie jest możliwe zmodyfikowanie odwzorowań istniejących ani utworzenie nowych.

Procedura:

Aby utworzyć odwzorowanie typu danych, należy użyć instrukcji CREATE TYPE MAPPING.

Pojęcia pokrewne:

- “Odwzorowania typów danych w systemie stowarzyszonym” na stronie 47

Informacje pokrewne:

- Rozdział 27, “Domyślne proste odwzorowania typów”, na stronie 269
- Rozdział 28, “Domyślne zwrotne odwzorowania typów”, na stronie 285
- “Tworzenie odwzorowania dla typu źródła danych – przykład” na stronie 51
- “Tworzenie odwzorowania dla typu danych i wersji źródła danych – przykład” na stronie 52
- “Tworzenie odwzorowania typów dla wszystkich obiektów źródła danych na serwerze – przykład” na stronie 53
- “Zmiana typu lokalnego dla obiektu źródła danych – przykłady” na stronie 55
- “Zmiany długich typów danych na typ varchar” na stronie 57
- “Plan the data type mappings” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Tworzenie odwzorowania dla typu źródła danych – przykład

W tym przykładzie wszystkie tabele i widoki w programie Oracle, których typem danych jest Oracle NUMBER, muszą być odwzorowane na typ DB2 DECIMAL(8,2). Typ danych Oracle NUMBER jest domyślnie odwzorowany na typ DB2 DOUBLE, czyli zmiennopozycyjne liczby dziesiętne.

Do zmiany typów lokalnych istniejących pseudonimów należy użyć instrukcji ALTER NICKNAME. Aby zmienić lokalny typ danych na DECIMAL(8,2), należy osobno zmodyfikować każdy pseudonim.

Jeśli pseudonimy nie istnieją, należy utworzyć odwzorowanie typu danych, określające typ źródła danych.

Na przykład aby utworzyć odwzorowanie z typu Oracle NUMBER na typ DB2 DECIMAL(8,2), użyj następującej instrukcji:

```
CREATE TYPE MAPPING MY_ORACLE_DEC FROM SYSIBM.DECIMAL(8,2)
    TO SERVER TYPE ORACLE TYPE NUMBER
```

MY_ORACLE_DEC

Nazwa nadawana odwzorowaniu typów. Nazwa nie może być identyczna z nazwą innego odwzorowania, które już istnieje w katalogu.

FROM *SYSIBM.DECIMAL(8,2)*

Lokalny schemat DB2 i lokalny typ danych. Jeśli długość lub precyzja i skala nie zostaną określone, wartości te są określane na podstawie typu w źródle danych.

TO SERVER TYPE *ORACLE*

Określa typ źródła danych.

TYPE *NUMBER*

Typ danych w źródle danych odwzorowywany na lokalny typ danych. Typy danych zdefiniowane przez użytkownika są niedozwolone.

Kolumnom Oracle zostaje lokalnie przypisany typ danych DB2 DECIMAL(8,2).

Wszystkie nowe tabele i widoki Oracle, zawierające kolumny typu NUMBER, również będą podczas tworzenia pseudonimów podlegały odwzorowaniu typu Oracle NUMBER na typ DB2 DECIMAL(8,2).

Zadania pokrewne:

- “Tworzenie odwzorowań typów danych” na stronie 51

Informacje pokrewne:

- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE TYPE MAPPING statement” w podręczniku *SQL Reference, Volume 2*

Tworzenie odwzorowania dla typu danych i wersji źródła danych – przykład

Ten przykład dotyczy tabel i widoków bazy danych Oracle na różnych wersjach serwera Oracle. Dla wszystkich tabel i widoków znajdujących się na serwerach Oracle, wersja 8.0.3, kolumny typu Oracle NUMBER(23,3) muszą być odwzorowane na typ DB2 DECIMAL(8,2). Typ danych Oracle NUMBER(23,3) jest domyślnie odwzorowywany na typ DB2 DECIMAL(23,3).

Do zmiany typów lokalnych istniejących pseudonimów należy użyć instrukcji ALTER NICKNAME. Aby zmienić lokalny typ danych na DECIMAL(8,2), należy osobno zmodyfikować każdy pseudonim.

Jeśli pseudonimy nie istnieją, należy utworzyć odwzorowanie typu danych, określające typ źródła danych.

Na przykład aby odwzorować typ danych Oracle NUMBER(23,3) na typ danych DB2 DECIMAL(8,2) dla serwerów Oracle w wersji 8.0.3, należy użyć następującej instrukcji:

```
CREATE TYPE MAPPING ORA_DEC FROM SYSIBM.DECIMAL(8,2)
    TO SERVER TYPE ORACLE VERSION 8.0.3 TYPE NUMBER(23,3)
```


ORA_DEC

Nazwa nadawana odwzorowaniu typów. Nazwa nie może być identyczna z nazwą innego odwzorowania, które już istnieje w katalogu.

FROM *SYSIBM.DECIMAL(8,2)*

Lokalny schemat DB2 i lokalny typ danych. Jeśli długość lub precyzja i skala nie zostaną określone, wartości te są określane na podstawie typu w źródle danych.

TO SERVER TYPE *ORACLE*

Określa typ źródła danych.

VERSION *8.0.3*

Określa wersję serwera źródła danych. Określenie wersji jest wymagane. Można także określić wydanie i numer modyfikacji, jak w podanym przykładzie.

TYPE *NUMBER(23,3)*

Typ danych w źródle danych odwzorowywany na lokalny typ danych. Typy danych zdefiniowane przez użytkownika są niedozwolone.

Dla kolumn Oracle na serwerze w wersji 8.0.3 program DB2 UDB definiuje lokalny typ danych DB2 DECIMAL(8,2).

Wszystkie nowe tabele i widoki Oracle na serwerze Oracle, wersja 8.0.3, zawierające kolumny typu NUMBER(23,3), będą podczas tworzenia pseudonimów także podlegały odwzorowaniu typu Oracle NUMBER(23,3) na typ DB2 DECIMAL(8,2).

Tabele i widoki Oracle na serwerach w wersji innej niż 8.0.3 korzystają z domyślnego odwzorowania typów.

Zadania pokrewne:

- “Tworzenie odwzorowań typów danych” na stronie 51

Informacje pokrewne:

- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE TYPE MAPPING statement” w podręczniku *SQL Reference, Volume 2*

Tworzenie odwzorowania typów dla wszystkich obiektów źródła danych na serwerze – przykład

W tym przykładzie w stowarzyszonej bazie danych definiowany jest serwer źródła danych ORA2SERVER. Każda z tabel zawiera kolumnę należącą do typu danych Oracle DATE. Wartości typu danych DATE zawierają zapis wieku, roku, miesiąca, dnia, godziny, minuty i sekundy. Typ danych Oracle DATE jest domyślnie odwzorowany na lokalny typ danych DB2 TIMESTAMP. Kiedy jednak uruchamia się zapytania dla dowolnego obiektu na tym serwerze, tabela wynikowa powinna zwracać tylko informacje o godzinie (pełną godzinę, minuty i sekundy).

Do zmiany typów lokalnych istniejących pseudonimów należy użyć instrukcji ALTER NICKNAME. Aby zmienić lokalny typ danych na TIME, należy osobno zmodyfikować każdy pseudonim.

Jeśli pseudonimy nie istnieją, należy utworzyć odwzorowanie typu danych, określające typ źródła danych.

Aby odwzorować typy danych Oracle DATE na typ DB2 TIME dla serwera ORA2SERVER, należy użyć następującej instrukcji:

```
CREATE TYPE MAPPING ORA2_DATE FROM SYSIBM.TIME
TO SERVER ORA2SERVER TYPE DATE
```

ORA2_DATE

Nazwa nadawana odwzorowaniu typów. Nazwa nie może być identyczna z nazwą innego odwzorowania, które już istnieje w katalogu.

FROM SYSIBM.TIME

Lokalny schemat DB2 i lokalny typ danych. Jeśli długość lub precyzja i skala nie zostaną określone, wartości te są określane na podstawie typu w źródle danych.

TO SERVER ORA2SERVER

Lokalna nazwa serwera źródła danych.

TYPE DATE

Typ danych w źródle danych odwzorowywany na lokalny typ danych. Typy danych zdefiniowane przez użytkownika są niedozwolone.

Dla kolumn Oracle typu DATE program DB2 UDB definiuje lokalnie typ DB2 TIME.

Każdy dodawany na serwerze nowy obiekt z kolumną typu DATE także będzie podlegał odwzorowaniu typu Oracle DATE na typ DB2 TIME podczas tworzenia odpowiadającego mu pseudonimu.

To odwzorowanie typu danych nie będzie miało zastosowania do obiektów źródeł danych na innych serwerach Oracle.

Zadania pokrewne:

- “Tworzenie odwzorowań typów danych” na stronie 51

Informacje pokrewne:

- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE TYPE MAPPING statement” w podręczniku *SQL Reference, Volume 2*

Zmiana typu lokalnego dla obiektu źródła danych

Po utworzeniu pseudonimu typy danych skojarzone z obiektem źródła danych są przechowywane w stowarzyszonej bazie danych. W przypadku niektórych źródeł danych lokalne typy danych są określane przez opakowanie. W przypadku innych źródeł danych konieczne jest określenie typów danych samodzielnie podczas tworzenia pseudonimu.

Można określić typ lokalny dla kolumny konkretnego obiektu źródła danych. Zamiast instrukcji CREATE TYPE MAPPING używa się wtedy instrukcji ALTER NICKNAME.

Uwaga: Zmiana lokalnego typu danych może spowodować błędy lub utratę informacji, jeśli lokalny typ danych dla kolumny zmieni się na typ, który znacznie różni się od typu zdalnego.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika uruchamiającego instrukcję musi mieć przyznane co najmniej jedno z następujących uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie ALTER względem pseudonimu występującego w instrukcji
- uprawnienie CONTROL względem pseudonimu występującego w instrukcji
- uprawnienie ALTERIN względem schematu, o ile nazwa schematu pseudonimu istnieje

Identyfikator autoryzowanego użytkownika skojarzony z instrukcją musi być jednocześnie identyfikatorem użytkownika definiującego pseudonim, który został zarejestrowany w kolumnie DEFINER widoku katalogu dla pseudonimu.

Ograniczenia:

Zobacz temat poświęcony ograniczenia dotyczące definiowania pseudonimów.

Procedura:

Typy danych można zmieniać z Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Wybierz folder **Pseudonimy**.
2. Kliknij prawym przyciskiem myszy pseudonim, który chcesz zmienić, i kliknij przycisk **Zmień**. Zostanie otwarty notatnik Zmodyfikuj pseudonim.
3. Na stronie Pseudonimy zaznacz kolumnę, którą chcesz zmienić, i kliknij przycisk **Zmień**. Zostanie otwarte okno Zmień kolumnę.
4. Wybierz typ danych.
5. Kliknij przycisk **OK**, aby zmienić typ danych i zamknąć okno.
6. Kliknij przycisk **OK**, aby zmodyfikować pseudonim i zamknąć notatnik.

Aby wykonać to zadanie z wiersza komend, należy użyć instrukcji ALTER NICKNAME. Na przykład:

```
ALTER NICKNAME pseudonim  
ALTER COLUMN nazwa_kolumny  
LOCAL TYPE typ_danych
```

Aby zawartość kolumny lokalnej typu znakowego traktować jako dane bitowe (binarne), należy w instrukcji ALTER NICKNAME użyć klauzuli FOR BIT DATA. Gdy użyje się tej klauzuli do modyfikacji lokalnego typu danych kolumny, podczas wymiany danych z innymi systemami nie będą wykonywane konwersje strony kodowej. Porównania są przeprowadzane binarnie, bez względu na kolejność zestawiania obowiązującą w zdalnej bazie danych.

Zadania pokrewne:

- “Modyfikowanie pseudonimu” na stronie 34

Informacje pokrewne:

- “Ograniczenia dotyczące zmiany pseudonimów” na stronie 35
- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*
- Rozdział 27, “Domyślne proste odwzorowania typów”, na stronie 269
- “Zmiana typu lokalnego dla obiektu źródła danych – przykłady” na stronie 55
- Rozdział 30, “Typy danych obsługiwane dla nierelacyjnych źródeł danych”, na stronie 301

Zmiana typu lokalnego dla obiektu źródła danych – przykłady

Poniższe przykłady przedstawiają sposób zmiany typów danych dla obiektu źródła danych.

Przykład: Odzworowanie liczbowego typu danych:

W tabeli Oracle z informacjami o pracownikach kolumna BONUS (Premia) jest zdefiniowana jako kolumna z typem danych NUMBER(32,3). Typ danych Oracle NUMBER(32,3) jest domyślnie odzworowany na typ DB2 DOUBLE, czyli liczby zmiennopozycyjne o podwójnej precyzji. Ze względu na to odzworowanie zapytanie dotyczące kolumny BONUS mogłoby zwracać wartości w następującej postaci:

```
5,00000000000000E+002  
1,00000000000000E+003
```

Zapis wykładniczy informuje o liczbie miejsc dziesiętnych oraz o kierunku, w którym należy przesunąć separator dziesiętny. W tym przykładzie wykładnik +002 oznacza, że separator dziesiętny należy przesunąć o dwa miejsca w prawą stronę, a wykładnik +003 oznacza, że separator należy przesunąć o trzy miejsca w prawo.

Zapytania operujące na kolumnie BONUS mogą zwracać wartości w formacie kwot pieniężnych. W tym celu wystarczy zmienić lokalną definicję kolumny BONUS dla tej tabeli z typu DOUBLE na typ DECIMAL. Warto przy tym podać precyzję i skalę właściwe dla zapisu kwot premii. Jeśli na przykład wiadomo, że całkowita część premii nigdy nie przekroczy kwoty sześciocyfrowej, typ NUMBER(32,3) można odzworować na typ DECIMAL(8,2). Ze względu na ograniczenia narzucone przez nowe odzworowanie, zapytania operujące na kolumnie BONUS będą zwracały wartości w postaci:

```
500,00  
1000,00
```

Pseudonim tabeli Oracle to ORASALES. Aby odzworować kolumnę BONUS w tabeli ORASALES na typ danych DB2 DECIMAL (8,2), należy użyć następującej instrukcji ALTER NICKNAME:

```
ALTER NICKNAME ORASALES ALTER COLUMN BONUS  
LOCAL TYPE DECIMAL(8,2)
```

ORASALES

Pseudonim zdefiniowany dla tabeli Oracle.

ALTER COLUMN BONUS

Nazwa kolumny zdefiniowana lokalnie w widoku katalogu SYSCAT.COLUMNS stowarzyszonej bazy danych.

LOCAL TYPE DECIMAL(8,2)

Określa nowy lokalny typ dla kolumny.

Odzworowanie to dotyczy tylko kolumny BONUS w tabeli Oracle wskazanej pseudonimem ORASALES. Wszystkie inne obiekty w źródle danych Oracle, które zawierają kolumnę BONUS, będą podlegały domyślnemu odzworowaniu typu danych Oracle NUMBER.

Przykład: Odzworowanie typu danych daty:

Pseudonim tabeli Oracle o nazwie SALES to ORASALES. Tabela SALES zawiera kolumnę o typie Oracle DATE. Domyślnie typ danych Oracle DATE jest odzworowany na typ DB2 TIMESTAMP. Jednak podczas pobierania danych z tej kolumny wymagane jest tylko wyświetlanie samej daty. W tym celu można zmodyfikować pseudonim tabeli SALES, aby zmienić typ lokalny na typ DB2 DATE.

```
ALTER NICKNAME ORASALES ALTER COLUMN ORDER_DATE  
LOCAL TYPE DATE
```

Przykład: Odzworowanie typu danych dla nierelacyjnego źródła danych:

Pseudonim dla pliku o strukturze tabeli o nazwie drugdata1.txt to DRUGDATA1. Plik drugdata1.txt zawiera kolumnę z nazwami leków. Nazwa tej kolumny to DRUG. Według oryginalnej definicji kolumna DRUG jest typu CHAR(20). Długość tej kolumny musi być zmieniona na CHAR(30). Aby zmienić odzworowanie na typ o odpowiedniej długości, można zmodyfikować pseudonim pliku drugdata1.txt:

```
ALTER NICKNAME DRUGDATA1 ALTER COLUMN DRUG  
LOCAL TYPE CHAR(30)
```

Zadania pokrewne:

- “Tworzenie odzworowań typów danych” na stronie 51
- “Zmiana typu lokalnego dla obiektu źródła danych” na stronie 54

Informacje pokrewne:

- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*
- “Ograniczenia dotyczące zmiany pseudonimów” na stronie 35

Zmiany długich typów danych na typy varchar

Aby umożliwić wykonywanie operacji wstawiania i aktualizowania dla długich typów danych, można zmienić te typy na typ VARCHAR. Tabela 4 przedstawia długie typy danych, które można zmienić, pogrupowane według źródeł danych.

Tabela 4. Długie typy danych, które można zmienić na typ varchar, pogrupowane według źródeł danych

Źródło danych	Zdalny typ danych	Długość	Lokalny domyślny typ danych	Instrukcja ALTER dla typu VARCHAR
DRDA	BLOB	1–32672	BLOB	varchar z klauzulą FOR BIT DATA
	CLOB	1–32672	CLOB	varchar
	long varchar	1–32672	CLOB	varchar
	long varchar z klauzulą FOR BIT DATA	1–32672	BLOB	varchar z klauzulą FOR BIT DATA
Informix	byte	1–32672	BLOB	varchar z klauzulą FOR BIT DATA
	text	1–32672	CLOB	varchar
Microsoft SQL Server	image	1–32672 zmienne hosta; 1–8000 literały	BLOB	varchar z klauzulą FOR BIT DATA
	text	1–32672 zmienne hosta; 1–8000 literały	CLOB	varchar
Oracle NET8	long	1–32672 zmienne hosta; 1–4000 literały	CLOB	varchar
	long raw	1–32672 zmienne hosta; 1–4000 literały	BLOB	varchar z klauzulą FOR BIT DATA

Tabela 4. Długie typy danych, które można zmienić na typ varchar, pogrupowane według źródeł danych (kontynuacja)

Źródło danych	Zdalny typ danych	Długość	Lokalny domyślny typ danych	Instrukcja ALTER dla typu VARCHAR
Sybase CTLIB	image	1–32672	BLOB	varchar z klauzulą FOR BIT DATA
	text	1–32672	CLOB	varchar
Teradata	byte	32673–64000	BLOB	varchar z klauzulą FOR BIT DATA (32672)
	CHAR	32673–64000	CLOB	varchar(32672)
	varbyte	32673–64000	BLOB	varchar z klauzulą FOR BIT DATA (32672)
	varchar	32673–64000	CLOB	varchar(32672)

Pojęcia pokrewne:

- “Odzworowania typów danych w systemie stowarzyszonym” na stronie 47
- “Instrukcje INSERT, UPDATE i DELETE a obiekty LOB” na stronie 97

Informacje pokrewne:

- Rozdział 27, “Domyślne proste odzworowania typów”, na stronie 269

Rozdział 4. Odwzorowanie funkcji i funkcje zdefiniowane przez użytkownika

Opakowania dołączone do programu DB2 Information Integrator zawierają domyślne odwzorowania funkcji między źródłami danych a programem DB2 dla systemów Linux, UNIX i Windows.

- “Odwzorowania funkcji w systemie stowarzyszonym”
- “Jak działają odwzorowania funkcji w systemie stowarzyszonym” na stronie 60
- “Wymagania dotyczące odwzorowań funkcji zdefiniowanych przez użytkownika” na stronie 61
- “Szablony funkcji” na stronie 62
- “Tworzenie szablonów funkcji” na stronie 62
- “Przekazywanie informacji o narzucie w odwzorowaniu funkcji na użytek optymalizatora zapytań” na stronie 64
- “Określanie nazw funkcji w odwzorowaniu funkcji” na stronie 66
- “Jak utworzyć odwzorowanie funkcji” na stronie 66
- “Funkcje zdefiniowane przez użytkownika w aplikacjach” na stronie 70
- “Wyłączanie domyślnego odwzorowania funkcji” na stronie 70
- “Usuwanie odwzorowania funkcji zdefiniowanej przez użytkownika” na stronie 71

Odwzorowania funkcji w systemie stowarzyszonym

W programie DB2[®] Information Integrator zdefiniowane są domyślne odwzorowania między istniejącymi wbudowanymi funkcjami źródła danych a wbudowanymi funkcjami DB2. Aby serwer stowarzyszony mógł rozpoznać funkcję źródła danych, funkcja ta musi zostać odwzorowana na istniejącą funkcję w programie DB2 dla systemów Linux, UNIX[®] i Windows[®].

Domyślne odwzorowania funkcji są zdefiniowane w modułach opakowujących.

W przypadku nierelacyjnych źródeł danych nie można przesłonić istniejących odwzorowań funkcji ani utworzyć nowych odwzorowań.

Kiedy należy utworzyć własne odwzorowania funkcji

Jeśli dla potrzebnej funkcji źródła danych nie jest zdefiniowane domyślne odwzorowanie, można utworzyć nowe odwzorowanie funkcji. Przyczyną braku odwzorowania może być nieobecność właściwej funkcji w programie DB2 dla systemów Linux, UNIX i Windows.

Inną przyczyną braku odwzorowania może być fakt, że źródło danych zawiera funkcję podobną do funkcji DB2, jednak zwracającą odmienny wynik. Jeśli funkcja w źródle danych zwraca nawet nieznacznie różniące się wyniki lub wyniki różniące się dla niektórych zestawów danych wejściowych, opakowanie z reguły nie zawiera odwzorowań do takiej funkcji. Jeśli jednak istniejące rozbieżności wyników nie mają w danej sytuacji znaczenia, można samodzielnie utworzyć odwzorowanie między funkcjami. Utworzenie odwzorowania może mieć korzystny wpływ na wydajność.

Z odwzorowań funkcji należy korzystać, gdy:

- w źródle danych udostępniona zostaje nowa funkcja wbudowana;

- w źródle danych udostępniona zostaje nowa funkcja zdefiniowana przez użytkownika;
- odpowiednia funkcja DB2 nie istnieje;
- odpowiednia funkcja istnieje, lecz zwraca nieco inne wyniki, przy czym różnice te nie mają w danej sytuacji znaczenia.

Ustawienia dotyczące odwzorowań funkcji są dostępne w widoku katalogu SYSCAT.FUNCMAPPINGS.

Po utworzeniu odwzorowania funkcji możliwa jest sytuacja, w której wartości zwracane przez funkcję realizowaną w źródle danych będą odbiegały od wartości zwracanych przez odpowiadającą jej funkcję realizowaną w stowarzyszonej bazie danych DB2. Program DB2 Information Integrator użyje odwzorowania funkcji, lecz skutkiem może być błąd składniowy SQL lub nieoczekiwane wyniki.

Dlaczego odwzorowania funkcji są ważne

Odwzorowania funkcji należą do kilku istotnych czynników uwzględnianych w ramach analizy przekazywania do źródła wykonywanej przez optymalizator zapytań. Podejmując decyzję w kwestii doboru najlepszego planu dostępu przy realizacji zapytania, optymalizator zapytań rozpatruje możliwości wykonania określonego typu funkcji lub operacji SQL przez źródło danych. Jeśli funkcja jest pozbawiona odwzorowania, przekazanie jej wykonania do źródła danych nie będzie możliwe. Przekazywanie realizacji funkcji i innych operacji do źródła danych pozwala uzyskać wzrost wydajności.

W przypadku gdy w źródle danych obecna jest funkcja zbliżona do funkcji DB2, lecz zwracająca nieco inne wyniki, zdefiniowanie odwzorowania między tymi funkcjami może poprawić wydajność. Na przykład funkcja Informix[®] STDEV (odchylenie standardowe) daje inny wynik niż funkcja DB2 STDDEV dla niektórych zestawów danych wejściowych. Z tego powodu opakowanie źródeł Informix nie zawiera domyślnego odwzorowania między tymi funkcjami. Jeśli różnice wyników nie mają w danej sytuacji istotnego znaczenia, otwiera się droga do poprawy wydajności zapytań korzystających ze źródeł danych Informix i z funkcji DB2 STDDEV. Po utworzeniu odwzorowania między funkcją Informix STDEV a funkcją DB2 STDDEV optymalizator zapytań zyskuje możliwość zlecenia wykonania funkcji serwerowi źródła danych.

Pojęcia pokrewne:

- “Jak działają odwzorowania funkcji w systemie stowarzyszonym” na stronie 60
- “Wymagania dotyczące odwzorowań funkcji zdefiniowanych przez użytkownika” na stronie 61
- “Szablony funkcji” na stronie 62
- “Jak utworzyć odwzorowanie funkcji” na stronie 66

Zadania pokrewne:

- “Przekazywanie informacji o narzucie w odwzorowaniu funkcji na użytek optymalizatora zapytań” na stronie 64
- “Określanie nazw funkcji w odwzorowaniu funkcji” na stronie 66

Jak działają odwzorowania funkcji w systemie stowarzyszonym

Gdy serwer stowarzyszony odbierze żądanie realizacji zapytania odwołującego się do funkcji, sprawdzane są informacje o odwzorowaniach między funkcjami DB2[®] a funkcjami źródła danych. Obecność tych informacji sprawdzana jest w dwóch miejscach:

- W opakowaniu. Opakowanie źródła danych zawiera domyślne odwzorowania funkcji.

- W widoku katalogu SYSCAT.FUNCMAPPINGS. Widok ten zawiera wpisy tworzone w celu zastąpienia lub poszerzenia domyślnych odwzorowań funkcji zawartych w opakowaniu. Znajdują się tu ponadto nowe odwzorowania tworzone w razie braku odwzorowań domyślnych. Gdy do danej funkcji ma zastosowanie kilka odwzorowań, użyte zostanie najnowsze z nich (utworzone jako ostatnie).

Opcje odwzorowań funkcji pozwalają określić informacje o funkcji oraz o szacowanym koszcie związanym z uruchomieniem funkcji w źródle danych. W ramach opcji odwzorowań funkcji definiowane są takie dane, jak:

- nazwa funkcji w zdalnym źródle danych
- szacowana liczba instrukcji przetwarzanych przy pierwszym i ostatnim wywołaniu funkcji w źródle danych
- szacowana liczba operacji we/wy wykonywanych przy pierwszym i ostatnim wywołaniu funkcji w źródle danych
- szacowana liczba instrukcji przetwarzanych przy każdym wywołaniu funkcji w źródle danych

Tworząc odwzorowanie funkcji, określa się odpowiedniość między funkcją ze źródła danych a jej odpowiednikiem w stowarzyszonej bazie danych. W przypadku gdy odpowiednia funkcja DB2 nie istnieje, lub gdy zamiarem użytkownika jest wymuszenie na serwerze stowarzyszonym wywoływania funkcji w źródle danych, można utworzyć szablon funkcji, który będzie pełnił rolę lokalnego odpowiednika.

Pojęcia pokrewne:

- “Odwzorowania funkcji w systemie stowarzyszonym” na stronie 59
- “Wymagania dotyczące odwzorowań funkcji zdefiniowanych przez użytkownika” na stronie 61

Wymagania dotyczące odwzorowań funkcji zdefiniowanych przez użytkownika

Aby w systemie stowarzyszonym było możliwe wywoływanie funkcji zdefiniowanej przez użytkownika w źródle danych, stowarzyszona baza danych musi zawierać skojarzenie tej funkcji ze specyfikacją funkcji zapisaną w katalogu globalnym na serwerze stowarzyszonym.

Możliwość odwzorowania specyfikacji funkcji w stowarzyszonej bazie danych na zdalną funkcję w źródle danych obwarowana jest dwoma warunkami:

- W bazie danych DB2[®] UDB musi istnieć funkcja, której sygnatura odpowiada sygnaturze funkcji w źródle danych. *Sygnatura* obejmuje nazwę funkcji oraz jej parametry wejściowe. *Wzajemna odpowiedniość* sygnatur zachodzi, gdy spełnione są poniższe warunki:
 - sygnatury zawierają takie same nazwy i jednakowe liczby parametrów;
 - typ danych każdego z parametrów w jednej sygnaturze jest identyczny z typem danych odpowiedniego parametru w drugiej sygnaturze lub może być do tego typu przekształcony.
- Jeśli w bazie danych DB2 UDB nie ma funkcji o odpowiedniej sygnaturze, można zdefiniować szablon funkcji mający taką sygnaturę. Następnie wystarczy odwzorować szablon funkcji na funkcję w źródle danych, która ma być wywoływana.

Ustawienia dotyczące odwzorowań funkcji są dostępne w widoku katalogu SYSCAT.FUNCMAPPINGS.

Pojęcia pokrewne:

- “Odwzorowania funkcji w systemie stowarzyszonym” na stronie 59
- “Funkcje zdefiniowane przez użytkownika w aplikacjach” na stronie 70

Zadania pokrewne:

- “Usuwanie odwzorowania funkcji zdefiniowanej przez użytkownika” na stronie 71

Szablony funkcji

Serwer stowarzyszony rozpoznaje funkcję źródła danych na podstawie zdefiniowanego odwzorowania między nią a odpowiadającą jej funkcją DB2[®] w stowarzyszonej bazie danych.

W razie braku odpowiedniej lokalnej funkcji DB2 można utworzyć szablon funkcji pełniący jej rolę.

Szablon funkcji jest to funkcja DB2 tworzona wyłącznie w celu wymuszenia na serwerze stowarzyszonym wywołania funkcji w źródle danych. W przeciwieństwie do zwykłej funkcji, szablon funkcji jest pozbawiony kodu wykonywalnego. Gdy serwer stowarzyszony odbiera zapytania odwołujące się do szablonu funkcji, następuje wywołanie odpowiedniej funkcji w źródle danych.

Do tworzenia szablonu funkcji służy instrukcja CREATE FUNCTION z parametrem AS TEMPLATE.

Po utworzeniu szablonu funkcji należy zdefiniować odwzorowanie funkcji między szablonem a funkcją źródła danych. Odwzorowanie funkcji tworzy się za pomocą instrukcji CREATE FUNCTION MAPPING.

Pojęcia pokrewne:

- “Odwzorowania funkcji w systemie stowarzyszonym” na stronie 59

Zadania pokrewne:

- “Tworzenie szablonów funkcji” na stronie 62

Tworzenie szablonów funkcji

Serwer stowarzyszony rozpoznaje funkcję źródła danych na podstawie zdefiniowanego odwzorowania między nią a odpowiadającą jej funkcją w stowarzyszonej bazie danych. W razie braku odpowiedniej funkcji lokalnej można utworzyć szablon funkcji pełniący jej rolę.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika skojarzony z instrukcją musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna bądź niejawna nazwa schematu funkcji;
- uprawnienie CREATEIN względem schematu, jeśli nazwa schematu funkcji istnieje.

Ograniczenia:

Jeśli funkcja źródła danych wymaga podania parametrów wejściowych:

- Odpowiednik po stronie DB2 musi mieć tę samą liczbę parametrów, co funkcja w źródle danych.
- Typy danych parametrów wejściowych funkcji DB2 muszą być zgodne z typami danych odpowiednich parametrów funkcji w źródle danych. Nie są dozwolone parametry typu LONG VARCHAR, LONG VARGRAPHIC, DATALINK ani typu zdefiniowanego przez użytkownika.

Jeśli funkcja źródła danych nie ma parametrów wejściowych, odpowiadająca jej funkcja DB2 również nie może mieć żadnych parametrów wejściowych.

Procedura:

Do utworzenia szablonu funkcji służy instrukcja CREATE FUNCTION z parametrem AS TEMPLATE, na przykład:

```
CREATE FUNCTION BONUS ()  
  RETURNS DECIMAL(8,2)  
  AS TEMPLATE  
  DETERMINISTIC  
  NO EXTERNAL ACTION
```

BONUS ()

Nazwa nadawana szablonowi funkcji.

RETURNS *DECIMAL(8,2)*

Typ danych wyniku.

AS TEMPLATE

Oznacza, że jest to szablon funkcji, a nie funkcja.

DETERMINISTIC

Określa, że funkcja zawsze zwraca ten sam wynik dla określonego zbioru wartości argumentów.

NO EXTERNAL ACTION

Określa, że funkcja nie wywiera żadnego wpływu na obiekty, które nie są zarządzane przez menedżera bazy danych.

Konieczne jest określenie klauzul DETERMINISTIC oraz NO EXTERNAL ACTION, zgodnie z tym, czy sama funkcja jest deterministyczna i czy powoduje jakiegokolwiek działania zewnętrzne. W przeciwnym razie operacje SQL realizowane na podstawie tego szablonu funkcji zostaną poddane ograniczeniom.

Po utworzeniu szablonu funkcji należy zdefiniować odwzorowanie funkcji między szablonem a funkcją źródła danych. Do tworzenia odwzorowania funkcji służy instrukcja CREATE FUNCTION MAPPING, na przykład:

```
CREATE FUNCTION MAPPING MY_INFORMIX_FUN FOR BONUS ()  
  SERVER TYPE INFORMIX OPTIONS (REMOTE_NAME 'BONUS()')
```

MY_INFORMIX_FUN

Nazwa nadawana odwzorowaniu funkcji. Nazwa nie może być identyczna z nazwą odwzorowania funkcji, które już figuruje w katalogu globalnym stowarzyszonej bazy danych. Nazwa musi być unikalna.

FOR *BONUS* ()

Nazwa lokalnego szablonu funkcji DB2. Parametry wejściowe o odpowiednich typach danych należy ująć w nawiasy.

SERVER TYPE INFORMIX

Określa typ źródła danych zawierającego funkcję będącą celem odwzorowania.

OPTIONS (REMOTE_NAME 'BONUS()')

Opcja identyfikująca nazwę funkcji w zdalnym źródle danych, która jest odwzorowywana na lokalny szablon funkcji DB2.

Pojęcia pokrewne:

- “Odwzorowania funkcji w systemie stowarzyszonym” na stronie 59
- “Szablony funkcji” na stronie 62

Informacje pokrewne:

- “CREATE FUNCTION (Sourced or Template) statement” w podręczniku *SQL Reference, Volume 2*

Przekazywanie informacji o narzucie w odwzorowaniu funkcji na użytek optymalizatora zapytań

Gdy do kompilatora SQL w programie DB2 wprowadzane jest zapytanie zawierające funkcję, rolą optymalizatora zapytań jest ustalenie, czy wykonanie tej funkcji może być przekazane do źródła danych. Założmy, że w wyniku analizy przekazywania do źródła okazało się, że dana funkcja może być wykonywana zarówno w bazie DB2 UDB, jak i w zdalnym źródle danych.

Tworząc odwzorowanie funkcji, można w jego definicji zawrzeć istotne informacje dotyczące potencjalnego kosztu, lub inaczej *narzutu* związanego z wykonaniem funkcji w źródle danych. Na podstawie tych oszacowań optymalizator zapytań porównuje przewidywane koszty wykonania funkcji w bazie danych DB2 i w źródle danych.

Kierując się uzyskanymi wynikami, optymalizator zapytań DB2 może ustalić najlepszą strategię realizacji zapytania. Podczas przetwarzania żądania rozproszonego optymalizator ocenia alternatywne strategie dostępu do danych i szacuje związane z nimi narzuty, aby zdecydować o uruchomieniu funkcji DB2 lub funkcji źródła danych. Zastosowana będzie strategia o najniższym oczekiwanym narzucie.

Informacje o szacowanym narzucie podawane są w wywołaniu instrukcji CREATE FUNCTION MAPPING. Na przykład instrukcja może określać przewidywaną liczbę instrukcji wymaganych do wywołania funkcji w źródle danych. Może też to być informacja o przewidywanej liczbie operacji we/wy generowanych na każdy bajt argumentów przekazywanych tej funkcji. Odpowiednie wartości szacunkowe są zapisywane w katalogu globalnym i dostępne w widoku SYSCAT.FUNCMAPOPTIONS. Gdy w odwzorowaniu używana jest funkcja DB2 (zamiast funkcji źródła danych lub szablonu funkcji DB2), katalog globalny zawiera oszacowania narzutu związanego z wywołaniem funkcji DB2. Odpowiednie wartości zawiera widok SYSCAT.ROUTINES.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Procedura:

Aby zdefiniować informacje statystyczne w instrukcji CREATE FUNCTION MAPPING, należy posłużyć się opcjami odwzorowania funkcji. W poniższej tabeli znajduje się lista opcji odwzorowania funkcji służących do definiowania narzutu oraz przypisane im wartości domyślne.

Tabela 5. Opcje odwzorowania funkcji określające narzut funkcji

Opcja	Poprawne ustawienia	Ustawienie domyślne
INITIAL_INSTS	Szacowana liczba instrukcji przetwarzanych przy pierwszym i ostatnim wywołaniu funkcji w źródle danych.	'0'
INITIAL_IOS	Szacowana liczba operacji we/wy wykonywanych przy pierwszym i ostatnim wywołaniu funkcji w źródle danych.	'0'
INSTS_PER_ARGBYTE	Szacowana liczba przetwarzanych instrukcji dla każdego bajtu zbioru argumentów przekazywanych do funkcji źródła danych.	'0'
INSTS_PER_INVOC	Szacowana liczba instrukcji przetwarzanych przy każdym wywołaniu funkcji w źródle danych.	'450'
IOS_PER_ARGBYTE	Szacowana liczba operacji we/wy realizowanych dla każdego bajtu zbioru argumentów przekazywanych do funkcji źródła danych.	'0'
IOS_PER_INVOC	Szacowana liczba operacji we/wy realizowanych przy każdym wywołaniu funkcji w źródle danych.	'0'
PERCENT_ARGBYTES	Szacowana średnia część procentowa argumentu (w bajtach), którą funkcja źródła danych faktycznie odczyta.	'100'

Przykład: Opcja odwzorowania funkcji PERCENT_ARGBYTES:

Załóżmy, że tworzone jest odwzorowanie zdefiniowanej przez użytkownika funkcji US_DOLLAR w źródle danych Oracle na utworzoną w tym celu funkcję zdefiniowaną przez użytkownika w bazie danych DB2. Serwer źródła danych Oracle nosi nazwę ORACLE2. Funkcja zdefiniowana przez użytkownika w bazie DB2 nosi nazwę DOLLAR, a odwzorowanie funkcji zostanie nazwane ORACLE_DOLLAR. Za pomocą opcji PERCENT_ARGBYTES w definicji odwzorowania prześlemy optymalizatorowi dokładniejsze informacje o narzucie funkcji. Odpowiednia instrukcja SQL mogłaby wyglądać następująco:

```
CREATE FUNCTION MAPPING ORACLE_DOLLAR FOR DOLLAR()
  SERVER ORACLE2
  OPTIONS (REMOTE_NAME 'US_DOLLAR()', PERCENT_ARGBYTES '250')
```

Przykład: Opcja odwzorowania funkcji INSTS_PER_INVOC:

Załóżmy, że lokalna funkcja UCASE(CHAR) ma być odwzorowana na zdefiniowaną przez użytkownika funkcję Oracle o nazwie UPPERCASE. Funkcja Oracle znajduje się w źródle danych o nazwie ORACLE2. W definicji odwzorowania ma być zawarta informacja o szacowanej liczbie instrukcji wykonywanych przy każdym wywołaniu funkcji Oracle. Służy do tego poniższa składnia:

```
CREATE FUNCTION MAPPING MY_ORACLE_FUN4 FOR SYSFUN.UCASE(CHAR)
  SERVER ORACLE2
  OPTIONS (REMOTE_NAME 'UPPERCASE', INSTS_PER_INVOC '1000')
```

Modyfikowanie informacji o narzucie

Jeśli oszacowania narzutu ulegną zmianie, odpowiednie modyfikacje wartości można wprowadzić do katalogu globalnego. Aby zarejestrowanie nowych oszacowań dla funkcji w źródle danych było możliwe, należy najpierw usunąć lub wyłączyć odwzorowanie funkcji. Następnie wystarczy ponownie utworzyć odwzorowanie, podając właściwe wartości w instrukcji CREATE FUNCTION MAPPING. Nowe oszacowania znajdują się w widoku katalogu SYSCAT.FUNCMAPPINGS. Aby zarejestrować zmienione wartości dotyczące funkcji DB2, wystarczy bezpośrednio zmodyfikować zawartość widoku katalogu SYSSTAT.ROUTINES.

Pojęcia pokrewne:

- “Odwzorowania funkcji w systemie stowarzyszonym” na stronie 59

Informacje pokrewne:

- Rozdział 25, “Opcje odwzorowań funkcji w systemach stowarzyszonych”, na stronie 261

Określanie nazw funkcji w odwzorowaniu funkcji

Wartość wprowadzana w instrukcji CREATE FUNCTION MAPPING zależy od tego, czy odwzorowywane funkcje mają takie same, czy różne nazwy.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Odwzorowywanie funkcji o tych samych nazwach:

Odwzorowania można tworzyć między dwoma funkcjami (lub szablonem funkcji DB2 i funkcją źródła danych), które mają taką samą nazwę. Załóżmy na przykład, że należy odwzorować funkcję o nazwie MYFUN zdefiniowaną przez użytkownika w źródle danych Informix na zdefiniowaną przez użytkownika funkcję programu DB2 o nazwie TINA.MYFUN. Serwer źródła danych Informix nosi nazwę INFORMIX2. Składnia instrukcji powinna być następująca:

```
CREATE FUNCTION MAPPING FOR TINA.MYFUN(SYSTEM.INTEGER) SERVER INFORMIX2
```

Odwzorowywanie funkcji o różnych nazwach:

Tworząc odwzorowanie między dwoma funkcjami (lub szablonem funkcji programu DB2 a funkcją źródła danych), które mają różne nazwy, wykonaj następujące czynności:

- Przypisz nazwę funkcji programu DB2 lub szablonu funkcji parametrowi *nazwa_funkcji*.
- Przypisz nazwę funkcji źródła danych opcji odwzorowania funkcji o nazwie REMOTE_NAME. Wartość opcji REMOTE_NAME musi mieć mniej niż 255 znaków.

Załóżmy na przykład, że należy odwzorować zdefiniowaną przez użytkownika funkcję o nazwie UPPERCASE w źródle danych Oracle na funkcję programu DB2 UCASE(CHAR). Serwer źródła danych Oracle nosi nazwę ORACLE2. Dodatkowo należy podać szacowaną liczbę instrukcji przypadających na wywołanie funkcji UPPERCASE. Niech to odwzorowanie funkcji nazywa się ORACLE_UPPER. Składnia instrukcji powinna być następująca:

```
CREATE FUNCTION MAPPING ORACLE_UPPER FOR SYSFUN.UCASE(CHAR)
SERVER ORACLE2 OPTIONS
(REMOTE_NAME 'UPPERCASE', INSTS_PER_INVOC '1000')
```

Pojęcia pokrewne:

- “Odwzorowania funkcji w systemie stowarzyszonym” na stronie 59

Jak utworzyć odwzorowanie funkcji

Instrukcja CREATE FUNCTION MAPPING umożliwia utworzenie alternatywnego odwzorowania funkcji, które zastąpi odwzorowania domyślne. Wykaz utworzonych przez użytkownika alternatywnych odwzorowań funkcji znajduje się w widoku katalogu SYSCAT.FUNCMAPPINGS.

Za pomocą instrukcji CREATE FUNCTION MAPPING można ponadto zdefiniować opcje odwzorowania funkcji. Informacje o zdefiniowanych opcjach odwzorowania funkcji są obecne w widoku katalogu SYSCAT.FUNCMAPOPTIONS.

Za pomocą instrukcji CREATE FUNCTION MAPPING można:

- Utworzyć odwzorowanie funkcji dla wszystkich źródeł danych określonego typu, na przykład dla wszystkich źródeł danych Informix®.
- Utworzyć odwzorowanie funkcji dla wszystkich źródeł danych określonego typu i wersji, na przykład dla wszystkich źródeł danych Informix 9.
- Utworzyć odwzorowanie funkcji dla konkretnego serwera.
- Przekazać informacje statystyczne na temat odwzorowania funkcji na użytek optymalizatora.
- Wyłączyć domyślne odwzorowanie funkcji lub odwzorowanie funkcji zdefiniowane przez użytkownika.

Istnieje także możliwość użycia instrukcji CREATE FUNCTION MAPPING w Centrum sterowania DB2® lub w procesorze wiersza komend (CLP). Ponadto instrukcję CREATE FUNCTION MAPPING można umieścić w kodzie aplikacji. Centrum sterowania DB2 nie obsługuje tworzenia ani modyfikowania odwzorowań funkcji.

Zadania pokrewne:

- “Tworzenie odwzorowania funkcji dla określonego typu źródła danych” na stronie 67
- “Tworzenie odwzorowania funkcji dla określonego typu i wersji źródła danych” na stronie 68
- “Tworzenie odwzorowania funkcji dla wszystkich obiektów źródła danych na określonym serwerze” na stronie 69

Definiowanie szczegółów odwzorowań funkcji

Instrukcja CREATE FUNCTION MAPPING umożliwia odwzorowanie funkcji źródła danych na funkcję programu DB2 lub na szablon funkcji.

Tworzenie odwzorowania funkcji dla określonego typu źródła danych

W razie potrzeby można utworzyć odwzorowanie do funkcji dla wszystkich źródeł danych określonego typu.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Ograniczenia:

W przypadku źródeł nierelacyjnych istniejących odwzorowań funkcji nie można zastąpić ani utworzyć w ich miejsce nowych odwzorowań.

Procedura:

Żałóżmy, że należy utworzyć odwzorowanie szablonu funkcji DB2 na funkcję Oracle zdefiniowaną przez użytkownika, i że odwzorowanie to ma być aktywne dla wszystkich źródeł danych Oracle. Szablon nosi nazwę STATS i należy do schematu o nazwie NOVA.

Funkcja Oracle zdefiniowana przez użytkownika nosi nazwę STATISTICS i należy do schematu o nazwie STAR. Instrukcja CREATE FUNCTION MAPPING ma postać:

```
CREATE FUNCTION MAPPING MY_ORACLE_FUN1
  FOR NOVA.STATS ( DOUBLE, DOUBLE )
  SERVER TYPE ORACLE
  OPTIONS (REMOTE_NAME 'STAR.STATISTICS')
```

Pojęcia pokrewne:

- “Jak utworzyć odwzorowanie funkcji” na stronie 66

Zadania pokrewne:

- “Tworzenie odwzorowania funkcji dla określonego typu i wersji źródła danych” na stronie 68
- “Tworzenie odwzorowania funkcji dla wszystkich obiektów źródła danych na określonym serwerze” na stronie 69
- “Usuwanie odwzorowania funkcji zdefiniowanej przez użytkownika” na stronie 71
- “Określanie nazw funkcji w odwzorowaniu funkcji” na stronie 66

Informacje pokrewne:

- “CREATE FUNCTION MAPPING statement” w podręczniku *SQL Reference, Volume 2*

Tworzenie odwzorowania funkcji dla określonego typu i wersji źródła danych

W razie potrzeby można utworzyć odwzorowanie funkcji dotyczące wszystkich źródeł danych o określonym typie i wersji.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Ograniczenia:

W przypadku źródeł nierelacyjnych istniejących odwzorowań funkcji nie można zastąpić ani utworzyć w ich miejsce nowych odwzorowań.

Procedura:

Załóżmy, że należy utworzyć odwzorowanie szablonu funkcji DB2 na funkcję Sybase zdefiniowaną przez użytkownika, i że odwzorowanie to ma być aktywne dla wszystkich źródeł danych Sybase na serwerach wersji 12. Szablon nosi nazwę SYB_STATS i należy do schematu o nazwie EARTH. Funkcja Sybase zdefiniowana przez użytkownika nosi nazwę STATISTICS i należy do schematu o nazwie MOON. Instrukcja CREATE FUNCTION MAPPING ma postać:

```
CREATE FUNCTION MAPPING SYBASE_STATS
  FOR EARTH.SYB_STATS ( DOUBLE, DOUBLE )
  SERVER TYPE SYBASE VERSION 12
  OPTIONS (REMOTE_NAME 'MOON.STATISTICS')
```

Pojęcia pokrewne:

- “Jak utworzyć odwzorowanie funkcji” na stronie 66

Zadania pokrewne:

- “Tworzenie odwzorowania funkcji dla określonego typu źródła danych” na stronie 67
- “Tworzenie odwzorowania funkcji dla wszystkich obiektów źródła danych na określonym serwerze” na stronie 69
- “Usuwanie odwzorowania funkcji zdefiniowanej przez użytkownika” na stronie 71
- “Określanie nazw funkcji w odwzorowaniu funkcji” na stronie 66

Informacje pokrewne:

- “CREATE FUNCTION MAPPING statement” w podręczniku *SQL Reference, Volume 2*

Tworzenie odwzorowania funkcji dla wszystkich obiektów źródła danych na określonym serwerze

W razie potrzeby można utworzyć odwzorowanie do funkcji, z którego korzystać będą wszystkie obiekty źródła danych na określonym serwerze zdalnym.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Ograniczenia:

W przypadku źródeł nierelacyjnych istniejących odwzorowań funkcji nie można zastąpić ani utworzyć w ich miejsce nowych odwzorowań.

Procedura:

Załóżmy, że szablon funkcji o nazwie BONUS ma zostać odwzorowany na funkcję zdefiniowaną przez użytkownika o nazwie BONUS. Przy tym odwzorowanie to powinno dotyczyć tylko serwera danych Oracle o nazwie ORA_SALES. Ponieważ nazwy funkcji są identyczne, nie ma potrzeby określania opcji odwzorowania funkcji REMOTE_NAME.

```
CREATE FUNCTION MAPPING BONUS_CALC FOR BONUS()  
SERVER ORA_SALES
```

Pojęcia pokrewne:

- “Jak utworzyć odwzorowanie funkcji” na stronie 66

Zadania pokrewne:

- “Tworzenie odwzorowania funkcji dla określonego typu źródła danych” na stronie 67
- “Tworzenie odwzorowania funkcji dla określonego typu i wersji źródła danych” na stronie 68
- “Usuwanie odwzorowania funkcji zdefiniowanej przez użytkownika” na stronie 71
- “Określanie nazw funkcji w odwzorowaniu funkcji” na stronie 66

Informacje pokrewne:

- “CREATE FUNCTION MAPPING statement” w podręczniku *SQL Reference, Volume 2*

Funkcje zdefiniowane przez użytkownika w aplikacjach

Programiści aplikacji często zmuszeni są do tworzenia własnego pakietu funkcji właściwego dla danej aplikacji lub dziedziny. Mogą używać w tym celu zdefiniowanych przez użytkownika funkcji skalarnych.

Na przykład w aplikacji przeznaczonej do obsługi handlu detalicznego można zdefiniować typ danych CENA przeznaczony do śledzenia kosztu sprzedawanych wyrobów. W tej aplikacji przydatne może być także zdefiniowanie funkcji PODATEK_VAT. Funkcja ta wyliczałaby wartość podatku VAT na podstawie ceny towaru i zwracała ją do użytkownika lub aplikacji.

Funkcje takie mogą wykonywać operacje na danych wszystkich typów dostępnych w bazie danych, w tym na dużych obiektach i danych typu odrębnego. Funkcje zdefiniowane przez użytkownika umożliwiają uwzględnianie w zapytaniach wszechstronnych obliczeń i predykatów wyszukiwania, które pozwalają odfiltrować niepotrzebne dane blisko źródła danych i tym samym skrócić czas odpowiedzi. Optymalizator SQL traktuje funkcje zdefiniowane przez użytkownika dokładnie tak samo, jak funkcje wbudowane, takie jak SUBSTR czy LENGTH. Do pisania aplikacji można używać różnych środowisk programistycznych, takich jak C, C++, COBOL i FORTRAN. Aplikacje mogą współużytkować zestaw zdefiniowanych przez użytkownika funkcji SQL, mimo, że funkcje te zostały napisane w różnych językach.

Funkcje zdefiniowane przez użytkownika mogą manipulować danymi i wykonywać różne działania. Można na przykład napisać funkcję, która będzie wysyłała wiadomość elektroniczną lub aktualizowała plik tekstowy.

W programie DB2[®] do funkcji zdefiniowanych przez użytkownika należą:

- funkcje zdefiniowane od podstaw,
- funkcje w schemacie SYSFUN; przykładami mogą być funkcje matematyczne, jak SIN, COS i TAN, funkcje naukowe, jak RADIANS, LOG10 i POWER, oraz funkcje ogólnego przeznaczenia, jak LEFT, DIFFERENCE i UCASE.

Pojęcia pokrewne:

- “User-defined functions” w podręczniku *SQL Reference, Volume 1*
- “Odzworowania funkcji w systemie stowarzyszonym” na stronie 59
- “Wymagania dotyczące odzworowań funkcji zdefiniowanych przez użytkownika” na stronie 61
- “Jak utworzyć odzworowanie funkcji” na stronie 66

Zadania pokrewne:

- “Usuwanie odzworowania funkcji zdefiniowanej przez użytkownika” na stronie 71

Wyłączanie domyślnego odzworowania funkcji

Domyślne odzworowania funkcji nie mogą być usuwane. Można jednak je wykluczyć z użytku.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Procedura:

Aby wyłączyć domyślne odwzorowanie funkcji, należy użyć instrukcji CREATE FUNCTION MAPPING, podając nazwę funkcji DB2 i ustawiając opcję DISABLE na 'Y'.

Załóżmy, że istnieje domyślne odwzorowanie funkcji między funkcją DB2 WEEK a zblizoną funkcją w źródle danych Oracle. Podczas realizacji zapytania pobierającego dane z serwera Oracle i wywołującego funkcję WEEK wywołanie może być skierowane do dowolnej z tych dwóch funkcji. To, która funkcja zostanie faktycznie wywołana, zależy od oceny optymalizatora zapytań, który szacuje narzut generowany przez każdą z nich.

Chcemy ocenić, jaki wpływ na wydajność będzie miało bezwarunkowe wywoływanie funkcji WEEK w źródle danych Oracle. Aby mieć pewność, że funkcja WEEK będzie zawsze uruchamiana na serwerze Oracle, należy wyłączyć domyślne odwzorowanie funkcji. Służy do tego poniższa składnia:

```
CREATE FUNCTION MAPPING FOR SYSFUN.WEEK(INT)
TYPE ORACLE OPTIONS (DISABLE 'Y')
```

Pojęcia pokrewne:

- “Odwzorowania funkcji w systemie stowarzyszonym” na stronie 59

Zadania pokrewne:

- “Usuwanie odwzorowania funkcji zdefiniowanej przez użytkownika” na stronie 71

Informacje pokrewne:

- “CREATE FUNCTION MAPPING statement” w podręczniku *SQL Reference, Volume 2*

Usuwanie odwzorowania funkcji zdefiniowanej przez użytkownika

Kiedy utworzone przez użytkownika odwzorowanie funkcji przestaje być potrzebne, można je usunąć.

Usunięcie odwzorowania funkcji zdefiniowanej przez użytkownika, które utworzono w celu zastąpienia domyślnego odwzorowania funkcji sprawi, że ponownie używane będzie odwzorowanie domyślne.

Lista odwzorowań funkcji zdefiniowanej przez użytkownika znajduje się w widoku katalogu SYSCAT.FUNCMAPPINGS.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane uprawnienie SYSADM lub DBADM.

Procedura:

Aby usunąć samodzielnie utworzone odwzorowanie funkcji, należy posłużyć się instrukcją DROP FUNCTION MAPPING.

Załóżmy, że zdefiniowane jest odwzorowanie o nazwie BONUS_CALC. Aby je usunąć, należy użyć funkcji DROP FUNCTION MAPPING w postaci:

```
DROP FUNCTION MAPPING BONUS_CALC
```

Pojęcia pokrewne:

- “Jak utworzyć odwzorowanie funkcji” na stronie 66

Zadania pokrewne:

- “Wyłączanie domyślnego odwzorowania funkcji” na stronie 70

Informacje pokrewne:

- “DROP statement” w podręczniku *SQL Reference, Volume 2*

Rozdział 5. Specyfikacje indeksu

Specyfikacja indeksu to zestaw metadanych dodawany do katalogu globalnego podczas tworzenia pseudonimu dla obiektu źródła danych. Optymalizator zapytań wykorzystuje te informacje do przyspieszenia przetwarzania żądań rozproszonych. W niektórych sytuacjach podczas tworzenia pseudonimu dla obiektu relacyjnego źródła danych specyfikacja indeksu nie jest tworzona. Należy wówczas samodzielnie utworzyć tę specyfikację.

W tym rozdziale opisano następujące zagadnienia:

- “Specyfikacje indeksów w systemie stowarzyszonym”
- “Tworzenie specyfikacji indeksów dla obiektów źródła danych” na stronie 74
- “Tworzenie specyfikacji indeksów dla tabel, do których dodano nowe indeksy” na stronie 75
- “Tworzenie specyfikacji indeksów dla widoków” na stronie 77
- “Tworzenie specyfikacji indeksów dla synonimów Informix” na stronie 78

Specyfikacje indeksów w systemie stowarzyszonym

W celu wprowadzenia specyfikacji indeksu do katalogu globalnego używa się instrukcji CREATE INDEX wraz z pseudonimem. Jeśli tabela uzyskuje nowy indeks, utworzona instrukcja CREATE INDEX odwołuje się do pseudonimu odpowiadającego tabeli i będzie zawierać informacje o indeksie tabeli w źródle danych. W przypadku tworzenia pseudonimu dla widoku utworzona instrukcja CREATE INDEX odwołuje się do pseudonimu widoku i będzie zawierać informacje o indeksie tabeli bazowej widoku. Specyfikacja indeksu informuje serwer stowarzyszony o kolumnach, które tworzą zdalny indeks, i ich właściwościach dotyczących unikalności. Nie określa ona statystycznych właściwości indeksu, takich jak liczba unikalnych wartości klucza indeksu.

Jeśli zdalny indeks był zdefiniowany w momencie tworzenia pseudonimu, nie ma potrzeby wprowadzania jego specyfikacji.

Serwer stowarzyszony nie tworzy specyfikacji indeksu przy tworzeniu pseudonimu dla:

- tabel bez indeksów;
- widoków, którym w katalogu zdalnym z reguły nie są przypisane żadne informacje indeksowe;
- obiektów w źródle danych pozbawionym zdalnego katalogu, z którego serwer stowarzyszony mógłby uzyskać informacje o indeksach.

Załóżmy, że tabela zostaje uzupełniona o nowy indeks, dodany do tych, które były zdefiniowane w momencie tworzenia pseudonimu. Ponieważ dane o indeksach są przekazywane do katalogu globalnego tylko w chwili tworzenia pseudonimu, serwer stowarzyszony nie jest poinformowany o istnieniu nowego indeksu. Podobnie podczas tworzenia pseudonimu dla widoku serwer stowarzyszony nie dysponuje żadnymi informacjami na temat tabeli (oraz jej indeksów), na bazie której widok został wygenerowany. W takiej sytuacji można samodzielnie wpisać potrzebne informacje do katalogu globalnego. Można utworzyć specyfikację indeksu dla tabel pozbawionych indeksu. Specyfikacja indeksu informuje optymalizator zapytań, które kolumny w tabeli należy przeszukać w celu najszybszego odnalezienia danych.

Specyfikacji indeksów należy używać z relacyjnymi źródłami danych. Utworzenie indeksu dla nierelacyjnego źródła danych nie wpłynie dodatnio na wydajność.

Zadania pokrewne:

- “Tworzenie specyfikacji indeksów dla obiektów źródła danych” na stronie 74
- “Tworzenie specyfikacji indeksów dla tabel, do których dodano nowe indeksy” na stronie 75
- “Tworzenie specyfikacji indeksów dla widoków” na stronie 77
- “Tworzenie specyfikacji indeksów dla synonimów Informix” na stronie 78
- “Cechy pseudonimu wpływające na globalną optymalizację” na stronie 149

Tworzenie specyfikacji indeksów dla obiektów źródła danych

Podczas tworzenia pseudonimu dla tabeli w źródle danych serwer stowarzyszony wprowadza do katalogu globalnego informacje na temat indeksów skojarzonych z tą tabelą. Z informacji tych korzysta następnie optymalizator zapytań, by usprawnić realizację żądań rozproszonych. Zapis w katalogu ma postać zestawu metadanych i nosi nazwę *specyfikacji indeksu*.

Serwer stowarzyszony nie tworzy specyfikacji indeksu w następujących sytuacjach:

- Tabela, dla której tworzony jest pseudonim, nie ma indeksu.
- Tworzony pseudonim skojarzony jest z obiektem źródła danych, który nie zawiera indeksów, takim jak widok, synonim Informix, plik o strukturze tabeli, arkusz Excel, algorytm BLAST lub plik ze znacznikami XML.
- Zdalny indeks bazuje na kolumnie LOB.
- Indeks zdalny zawiera klucz o łącznej długości przekraczającej 1024 bajty.
- Maksymalna liczba części klucza jest większa niż 16.

W powyższych sytuacjach serwer stowarzyszony nie zapisuje specyfikacji indeksu dla obiektu źródła danych. Jednak w przypadku pierwszych dwóch pozycji na powyższej liście można samodzielnie wpisać do katalogu globalnego potrzebne informacje o indeksie. Do określenia informacji o indeksie można użyć instrukcji CREATE INDEX.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika skojarzony z instrukcją musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- jedno z uprawnień - CONTROL względem obiektu lub INDEX względem obiektu Oraz jedno z uprawnień - IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna lub niejawną nazwa schematu indeksu, lub uprawnienie CREATEIN względem schematu, jeśli nazwa schematu indeksu odwołuje się do istniejącego schematu.

Ograniczenia:

Podczas tworzenia specyfikacji indeksu dla pseudonimu obowiązują pewne ograniczenia.

- Jeśli ma zastosowanie opcja wiązania DYNAMICRULES BIND, nie jest możliwe dynamiczne przygotowanie instrukcji. Ponadto nie można korzystać z parametrów INCLUDE, CLUSTER, PCTFREE, MINPCTUSED, DISALLOW REVERSE SCANS i ALLOW REVERSE SCANS w instrukcji CREATE INDEX.
- Parametru UNIQUE można użyć tylko w przypadku, gdy dane tworzące klucz indeksu składają się z unikalnych wartości w każdym wierszu tabeli źródłowej. Faktyczna unikalność danych nie będzie sprawdzana.

- Suma nominalnych długości wskazanych kolumn nie może przekroczyć 1024.
- Indeks nie może zawierać żadnych kolumn typu LOB, DATALINK, ani kolumn typu odrębnego, który bazowałby na typie LOB lub DATALINK. Ograniczenie to jest egzekwowane nawet w przypadku, gdy długość kolumny nie przekracza limitu 1024 bajtów.

Składnia instrukcji CREATE INDEX:

Instrukcja CREATE INDEX może być osadzona w kodzie aplikacji lub wydana jako dynamiczna instrukcja SQL w wierszu komend lub w Centrum komend.

Instrukcja CREATE INDEX użyta wobec pseudonimu tworzy specyfikację indeksu w katalogu globalnym stowarzyszonej bazy danych. Nie jest przy tym tworzony rzeczywisty indeks tabeli w źródle danych.

Instrukcja tworząca specyfikację indeksu ma następującą postać:

```
CREATE INDEX nazwa_indeksu ON pseudonim
(nazwa_kolumny) SPECIFICATION ONLY
CREATE UNIQUE INDEX nazwa_indeksu ON pseudonim
(nazwa_kolumny DESC) SPECIFICATION ONLY
```

W specyfikacji indeksu *nazwa_kolumny* jest nazwą, za pomocą której serwer stowarzyszony odwołuje się do kolumny tabeli w źródle danych.

Pojęcia pokrewne:

- “Specyfikacje indeksów w systemie stowarzyszonym” na stronie 73

Zadania pokrewne:

- “Tworzenie specyfikacji indeksów dla tabel, do których dodano nowe indeksy” na stronie 75
- “Tworzenie specyfikacji indeksów dla widoków” na stronie 77
- “Tworzenie specyfikacji indeksów dla synonimów Informix” na stronie 78

Informacje pokrewne:

- “CREATE INDEX statement” w podręczniku *SQL Reference, Volume 2*

Tworzenie specyfikacji indeksów dla tabel, do których dodano nowe indeksy

Istnieje kilka sytuacji, w których tabela może uzyskać nowy indeks:

- Utworzono pseudonim dla tabeli początkowo pozbawionej indeksu, a następnie dodano indeks do tabeli.
- Utworzono pseudonim dla tabeli początkowo mającej jeden indeks, a następnie dodano do tabeli nowy indeks.

W powyższych sytuacjach należy utworzyć dla tabeli specyfikację indeksu. Z zawartych w niej informacji skorzysta kompilator SQL podczas przetwarzania zapytań odwołujących się do tej tabeli.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika skojarzony z instrukcją musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- jedno z uprawnień - CONTROL względem obiektu lub INDEX względem obiektu Oraz jedno z uprawnień - IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna lub niejawna nazwa schematu indeksu, lub uprawnienie CREATEIN względem schematu, jeśli nazwa schematu indeksu odwołuje się do istniejącego schematu.

Ograniczenia:

Podczas tworzenia indeksu dla pseudonimu obowiązują pewne ograniczenia.

- Jeśli ma zastosowanie opcja wiązania DYNAMICRULES BIND, nie jest możliwe dynamiczne przygotowanie instrukcji. Ponadto nie można korzystać z parametrów INCLUDE, CLUSTER, PCTFREE, MINPCTUSED, DISALLOW REVERSE SCANS i ALLOW REVERSE SCANS w instrukcji CREATE INDEX.
- Parametru UNIQUE można użyć tylko w przypadku, gdy dane tworzące klucz indeksu składają się z unikalnych wartości w każdym wierszu tabeli źródłowej. Faktyczna unikalność danych nie będzie sprawdzana.
- Suma nominalnych długości wskazanych kolumn nie może przekroczyć 1024.
- Indeks nie może zawierać żadnych kolumn typu LOB, DATALINK, ani kolumn typu odrębnego, który bazowałby na typie LOB lub DATALINK. Ograniczenie to jest egzekwowane nawet w przypadku, gdy długość kolumny nie przekracza limitu 1024 bajtów.

Przykład: Utworzono indeks dla tabeli początkowo go pozbawionej:

Załóżmy, że tworzony jest pseudonim *employee* związany z tabelą źródła danych o nazwie CURRENT_EMP, która nie ma indeksu. Jakiś czas po utworzeniu pseudonimu dla tabeli CURRENT_EMP zdefiniowany został indeks, w którym rolę klucza pełnią kolumny WORKDEPT i JOB.

Instrukcja tworząca specyfikację tego indeksu miałaby następującą postać:

```
CREATE UNIQUE INDEX job_by_dept ON employee
(WORKDEPT, JOB) SPECIFICATION ONLY
```

gdzie *job_by_dept* jest nazwą indeksu.

Przykład: Do tabeli zostaje dodany nowy indeks:

Załóżmy teraz, że utworzono pseudonim *jp_sales* dla tabeli o nazwie JAPAN_SALES. Następnie do tabeli dodany został nowy indeks, dołączając do indeksów obecnych już w chwili tworzenia pseudonimu. Nowy indeks w charakterze klucza używa kolumny MARKUP.

Instrukcja tworząca specyfikację tego indeksu miałaby następującą postać:

```
CREATE UNIQUE INDEX jp_markup ON jp_sales (MARKUP) SPECIFICATION ONLY
```

gdzie *jp_markup* jest nazwą indeksu.

Pojęcia pokrewne:

- “Specyfikacje indeksów w systemie stowarzyszonym” na stronie 73

Zadania pokrewne:

- “Tworzenie specyfikacji indeksów dla obiektów źródła danych” na stronie 74
- “Tworzenie specyfikacji indeksów dla widoków” na stronie 77
- “Tworzenie specyfikacji indeksów dla synonimów Informix” na stronie 78

Informacje pokrewne:

- “CREATE INDEX statement” w podręczniku *SQL Reference, Volume 2*

Tworzenie specyfikacji indeksów dla widoków

Podczas tworzenia pseudonimu dla widoku serwer stowarzyszony nie dysponuje żadnymi informacjami na temat tabeli (i jej indeksów), na bazie której widok został wygenerowany. Korzystne jest utworzenie dla widoku specyfikacji indeksu, której kompilator SQL będzie używał podczas przetwarzania zapytań odwołujących się do widoku.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika skojarzony z instrukcją musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- jedno z uprawnień - CONTROL względem obiektu lub INDEX względem obiektu Oraz jedno z uprawnień - IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna lub niejawna nazwa schematu indeksu, lub uprawnienie CREATEIN względem schematu, jeśli nazwa schematu indeksu odwołuje się do istniejącego schematu.

Ograniczenia:

Podczas tworzenia indeksu dla pseudonimu obowiązują pewne ograniczenia.

- Jeśli ma zastosowanie opcja wiązania DYNAMICRULES BIND, nie jest możliwe dynamiczne przygotowanie instrukcji. Ponadto nie można korzystać z parametrów INCLUDE, CLUSTER, PCTFREE, MINPCTUSED, DISALLOW REVERSE SCANS i ALLOW REVERSE SCANS w instrukcji CREATE INDEX.
- Parametru UNIQUE można użyć tylko w przypadku, gdy dane tworzące klucz indeksu składają się z unikalnych wartości w każdym wierszu tabeli źródłowej. Faktyczna unikalność danych nie będzie sprawdzana.
- Suma nominalnych długości wskazanych kolumn nie może przekroczyć 1024.
- Indeks nie może zawierać żadnych kolumn typu LOB, DATALINK, ani kolumn typu odrębnego, który bazowałby na typie LOB lub DATALINK. Ograniczenie to jest egzekwowane nawet w przypadku, gdy długość kolumny nie przekracza limitu 1024 bajtów.

Tworzenie pseudonimu dla widoku:

Załóżmy, że tworzony jest pseudonim *jp_sales2003* dla widoku o nazwie JAPAN_SALES2003. Tabelą bazową dla tego widoku jest tabela JAPAN_SALES zawierająca kilka indeksów: REGION, AMOUNT, SALES_REP. Utworzona instrukcja CREATE INDEX odwoła się do pseudonimu widoku i będzie zawierać informacje o indeksie tabeli bazowej widoku.

Tworząc specyfikację indeksu dla widoku, należy się upewnić, że kolumna lub kolumny, na podstawie których tworzony jest indeks, wchodzi w skład widoku. Jeśli wymagane jest utworzenie specyfikacji dla wszystkich indeksów tabeli bazowej, każda specyfikacja musi zostać utworzona niezależnie. Na przykład instrukcja tworząca specyfikację indeksu REGION miałaby następującą postać:

```
CREATE UNIQUE INDEX jp_2003_region ON jp_sales2003  
(REGION) SPECIFICATION ONLY
```

gdzie *jp_2003_region* jest nazwą indeksu, a *jp_sales2003* jest pseudonimem widoku JAPAN_SALES2003.

Pojęcia pokrewne:

- “Specyfikacje indeksów w systemie stowarzyszonym” na stronie 73

Zadania pokrewne:

- “Tworzenie specyfikacji indeksów dla obiektów źródła danych” na stronie 74
- “Tworzenie specyfikacji indeksów dla tabel, do których dodano nowe indeksy” na stronie 75
- “Tworzenie specyfikacji indeksów dla synonimów Informix” na stronie 78

Informacje pokrewne:

- “CREATE INDEX statement” w podręczniku *SQL Reference, Volume 2*

Tworzenie specyfikacji indeksów dla synonimów Informix

W programie Informix istnieje możliwość tworzenia synonimów dla tabel i widoków. Serwer stowarzyszony pozwala wprawdzie tworzyć pseudonimy odpowiadające synonimom Informix, jednak działanie programu zależy od tego, czy synonim dotyczy tabeli, czy widoku:

- Załóżmy, że tworzony jest pseudonim dla synonimu odpowiadającego tabeli Informix. Jeśli serwer stowarzyszony ustali, że tabela, z którą związany jest synonim, ma indeks, to dla synonimu tworzona jest specyfikacja indeksu. Jeśli tabela, do której odwołuje się synonim, nie ma indeksu, specyfikacja indeksu nie jest tworzona. Można jednak taką specyfikację utworzyć ręcznie za pomocą instrukcji CREATE INDEX.
- Załóżmy, że tworzony jest pseudonim dla synonimu odpowiadającego widokowi Informix. Serwer stowarzyszony nie dysponuje żadnymi informacjami na temat tabel, na podstawie których widok został utworzony. Z tego względu dla synonimu nie jest tworzona żadna specyfikacja indeksu. Można jednak taką specyfikację utworzyć ręcznie za pomocą instrukcji CREATE INDEX.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika skojarzony z instrukcją musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- jedno z uprawnień - CONTROL względem obiektu lub INDEX względem obiektu Oraz jedno z uprawnień - IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna lub niejawna nazwa schematu indeksu, lub uprawnienie CREATEIN względem schematu, jeśli nazwa schematu indeksu odwołuje się do istniejącego schematu.

Ograniczenia:

Podczas tworzenia indeksu dla pseudonimu obowiązują pewne ograniczenia.

- Jeśli ma zastosowanie opcja wiązania DYNAMICRULES BIND, nie jest możliwe dynamiczne przygotowanie instrukcji. Ponadto nie można korzystać z parametrów INCLUDE, CLUSTER, PCTFREE, MINPCTUSED, DISALLOW REVERSE SCANS i ALLOW REVERSE SCANS w instrukcji CREATE INDEX.
- Parametru UNIQUE można użyć tylko w przypadku, gdy dane tworzące klucz indeksu składają się z unikalnych wartości w każdym wierszu tabeli źródłowej. Faktyczna unikalność danych nie będzie sprawdzana.
- Suma nominalnych długości wskazanych kolumn nie może przekroczyć 1024.

- Indeks nie może zawierać żadnych kolumn typu LOB, DATALINK, ani kolumn typu odrębnego, który bazowałby na typie LOB lub DATALINK. Ograniczenie to jest egzekwowane nawet w przypadku, gdy długość kolumny nie przekracza limitu 1024 bajtów.

Przykład: Tworzenie pseudonimu dla synonimu Informix bazującego na tabeli:

Gdy synonim bazuje na tabeli Informix, która nie ma indeksu, można dla niego samodzielnie utworzyć specyfikację indeksu informującą optymalizator, które kolumny umożliwiają szybkie wyszukiwanie danych. Odpowiednia instrukcja określać będzie pseudonim synonimu oraz informacje o jednej lub kilku kolumnach w tabeli, do której synonim się odwołuje. Załóżmy, że tworzony jest pseudonim *contracts* dla synonimu o nazwie SALES_CONTRACTS, oraz że tabela, na której ten synonim bazuje, zawiera kilka indeksów: REGION, AMOUNT, SALES_REP. Utworzona instrukcja CREATE INDEX odwoła się do pseudonimu synonimu i będzie zawierać informacje o indeksie tabeli bazowej synonimu.

Przykład: Tworzenie pseudonimu dla synonimu Informix bazującego na widoku:

Założmy, że tworzony jest pseudonim *jp_sales2003* dla widoku o nazwie JAPAN_SALES2003. Tabelą bazową dla tego widoku jest tabela JAPAN_SALES zawierająca kilka indeksów: REGION, AMOUNT, SALES_REP. Utworzona instrukcja CREATE INDEX odwoła się do pseudonimu widoku i będzie zawierać informacje o indeksie tabeli bazowej widoku.

Tworząc specyfikację indeksu dla widoku należy się upewnić, że kolumna lub kolumny, na podstawie których tworzony jest indeks, wchodzi w skład widoku. Jeśli wymagane jest utworzenie specyfikacji dla wszystkich indeksów tabeli bazowej, każda specyfikacja musi zostać utworzona niezależnie.

Instrukcja tworząca specyfikację indeksu REGION miałaby następującą postać:
 CREATE UNIQUE INDEX jp_2003_region ON jp_sales2003 (REGION) SPECIFICATION ONLY

gdzie *jp_2003_region* jest nazwą indeksu, a *jp_sales2003* jest pseudonimem widoku JAPAN_SALES2003.

Pojęcia pokrewne:

- “Specyfikacje indeksów w systemie stowarzyszonym” na stronie 73

Zadania pokrewne:

- “Tworzenie specyfikacji indeksów dla obiektów źródła danych” na stronie 74
- “Tworzenie specyfikacji indeksów dla tabel, do których dodano nowe indeksy” na stronie 75
- “Tworzenie specyfikacji indeksów dla widoków” na stronie 77

Informacje pokrewne:

- “CREATE INDEX statement” w podręczniku *SQL Reference, Volume 2*

Rozdział 6. Przezroczysty kod DDL

W systemie stowarzyszonym możliwe jest utworzenie tabel w zdalnych źródłach danych za pośrednictwem serwera stowarzyszonego programu DB2 dla systemów Linux, UNIX i Windows. Tabele takie mogą być również modyfikowane i usuwane przez serwer stowarzyszony. Możliwość tworzenia zdalnych tabel przez stowarzyszony serwer programu DB2 nazywa się *przezroczystym kodem DDL*.

W niniejszym rozdziale opisano następujące zagadnienia:

- “Co to jest przezroczysty kod DDL?”
- “Zdalne kolumny LOB i przezroczysty kod DDL” na stronie 83
- “Tworzenie nowych tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 83
- “Modyfikowanie tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 86
- “Usuwanie tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 88

Co to jest przezroczysty kod DDL?

Przezroczysty kod DDL umożliwia tworzenie i modyfikowanie zdalnych tabel za pośrednictwem programu DB2[®] Information Integrator bez potrzeby posługiwania się sesjami tranzytowymi.

Instrukcje SQL używane w połączeniu z przezroczystym kodem DDL, to CREATE TABLE, ALTER TABLE i DROP TABLE.

Instrukcja CREATE TABLE z przezroczystym kodem DDL tworzy zdalną tabelę w źródle danych oraz odpowiadający jej pseudonim na serwerze stowarzyszonym. Podane typy danych programu DB2 zostaną przy tym odwzorowane na zdalne typy danych przy użyciu domyślnych ustawień zwrotnych odwzorowań typów. W przypadku większości źródeł danych domyślne odwzorowania typów są zdefiniowane w opakowaniach. Domyślne odwzorowania typów dla źródeł danych z rodziny DB2 określone są w opakowaniu DRDA[®]. Domyślne odwzorowania typów dla źródeł Informix[®] zawiera opakowanie INFORMIX, itd.

Korzyść ze stosowania przezroczystego kodu DDL polega na tym, że administratorzy baz danych DB2 mogą posługiwać się znanymi sobie procedurami do tworzenia tabel lokalnych i zdalnych. Przezroczysty kod DDL pozwala dzięki temu scentralizować administrowanie tabelami i ułatwia nadawanie uprawnień.

Przezroczysty kod DDL jest obsługiwany w przypadku następujących źródeł danych:

- DB2 for z/OS[™] and OS/390[®]
- DB2 for iSeries[™]
- DB2 dla systemów Linux, UNIX[®] i Windows[®]
- DB2 Server for VM and VSE
- Informix
- Microsoft[®] SQL Server
- ODBC
- Oracle
- Sybase
- Teradata

Przy tworzeniu tabel administrator bazy danych może korzystać z Centrum sterowania DB2 albo z instrukcji DDL w procesorze wiersza komend DB2 (CLP). Posługiwanie się przezroczystym kodem DDL pozwala uniknąć potrzeby opanowania różnych składni instrukcji DDL wymaganych dla różnych źródeł danych.

Aby utworzenie zdalnych tabel w źródle danych za pośrednictwem programu DB2 Information Integrator było możliwe, należy najpierw skonfigurować dostęp do źródła danych:

- w katalogu globalnym musi być zdefiniowane opakowanie dla odpowiedniego źródła danych;
- należy utworzyć definicję serwera, na którym znajduje się tabela zdalna;
- należy zdefiniować odwzorowanie użytkowników między bazą DB2 Information Integrator a serwerem źródła danych.

Do tworzenia zdalnych tabel warto używać kreatora tabel w Centrum sterowania DB2.

Identyfikator autoryzowanego użytkownika skojarzony z instrukcjami przezroczystego kodu DDL musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie CREATETAB względem bazy danych i uprawnienie USE względem obszaru tabel, jak również jedno z uprawnień:
 - IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna bądź niejawną nazwa schematu tabeli;
 - CREATEIN względem schematu, jeśli w instrukcji tworzenia tabeli użyto nazwy istniejącego schematu.

Aby wydawanie instrukcji przezroczystego kodu DDL było możliwe, niezbędne jest posiadanie przez dany identyfikator autoryzowanego użytkownika stosownych uprawnień względem pseudonimu (aby stowarzyszona baza danych przyjęła żądanie) oraz istnienie odpowiednich uprawnień na serwerze zdalnego źródła danych (aby żądanie zostało przyjęte przez źródło).

Ograniczenia w stosowaniu przezroczystego kodu DDL

Stosowanie przezroczystego kodu DDL podlega pewnym ograniczeniom:

- Nie można modyfikować ani usuwać tabel, które zostały utworzone w zdalnym źródle danych w trybie rodzimym.
- W zdalnych źródłach danych nie można tworzyć zmaterializowanych tabel zapytań.
- W definicji tabeli można określić podstawowe informacje o kolumnach, jednak nie można określić opcji tabel ani opcji kolumn. Na przykład nie są obsługiwane opcje LOB (LOGGED i COMPACT).
- Nie można określić komentarza dla kolumny.
- Nie można wygenerować zawartości kolumny.
- Można określić klucz podstawowy, ale nie można określić klucza obcego, klucza unikalnego ani ograniczeń sprawdzających. Kolumny użyte w charakterze klucza podstawowego muszą mieć przypisany atrybut NOT NULL i nie mogą zawierać danych typu LOB.
- Nie można modyfikować parametrów istniejących kolumn, takich jak typ lub długość danych.
- Nie jest obsługiwana klauzula DEFAULT w instrukcjach CREATE TABLE i ALTER TABLE.

Zadania pokrewne:

- “Tworzenie nowych tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 83
- “Modyfikowanie tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 86
- “Usuwanie tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 88

Zdalne kolumny LOB i przezroczysty kod DDL

Niektóre źródła danych, jak Oracle i Informix[®], nie przechowują informacji o długości kolumn LOB w katalogach systemowych. Przy tworzeniu pseudonimu dotyczącego tabeli pobierane są informacje z katalogu systemowego źródła danych, łącznie z długościami kolumn. Ponieważ dla kolumn typu LOB długość nie jest określona, w stowarzyszonej bazie danych przyjmowane jest założenie, że kolumna może mieć maksymalną dozwoloną długość dla typu LOB w programie DB2[®] dla systemów Linux, UNIX[®] i Windows[®]. Dlatego w katalogu stowarzyszonej bazy danych na serwerze DB2 dla systemów Linux, UNIX i Windows długość kolumny dla pseudonimu zostaje zapisana jako długość maksymalna.

Jednak podczas tworzenia zdalnej tabeli przy użyciu przezroczystych instrukcji DDL zdefiniowanie długości kolumny LOB jest niezbędne. Kiedy serwer stowarzyszony tworzy pseudonim skojarzony ze zdalną tabelą, podana długość jest zapisywana w katalogu stowarzyszonej bazy danych jako długość kolumny pseudonimu.

Pojęcia pokrewne:

- “Co to jest przezroczysty kod DDL?” na stronie 81
- “Stowarzyszona obsługa obiektów LOB” na stronie 208

Zadania pokrewne:

- “Tworzenie nowych tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 83

Tworzenie nowych tabel zdalnych za pomocą przezroczystego kodu DDL

Aby utworzyć zdalną tabelę przy użyciu przezroczystego kodu DDL, można posłużyć się Centrum sterowania DB2 lub instrukcją CREATE TABLE.

Kreator zdalnych tabel w Centrum sterowania DB2 eliminuje ryzyko użycia nieobsługiwanego parametru lub opcji. Za pomocą kreatora można określić kolumny, wybierając je z listy predefiniowanych kolumn lub definiując atrybuty nowej kolumny.

Wymagania wstępne:

Aby utworzenie zdalnej tabeli było możliwe, należy skonfigurować serwer stowarzyszony pod kątem dostępu do źródeł danych. Konfiguracja obejmuje następujące etapy:

- utworzenie opakowania dla odpowiedniego typu źródła danych;
- utworzenie definicji serwera, na którym zdalne źródło danych się znajduje;
- utworzenie odwzorowań użytkowników między programem DB2 Information Integrator a serwerem zdalnego źródła danych.

Aby wydawanie instrukcji przezroczystego kodu DDL było możliwe, niezbędne jest posiadanie przez dany identyfikator autoryzowanego użytkownika stosownych uprawnień względem pseudonimu (aby stowarzyszona baza danych przyjęła żądanie) oraz istnienie odpowiednich uprawnień na serwerze zdalnego źródła danych (aby żądanie zostało przyjęte przez źródło).

Identyfikator autoryzowanego użytkownika skojarzony z przezroczystymi instrukcjami DDL musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie CREATETAB względem bazy danych i uprawnienie USE względem obszaru tabel, jak również jedno z uprawnień:
 - IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna bądź niejawną nazwa schematu tabeli;
 - CREATEIN względem schematu, jeśli w instrukcji tworzenia tabeli użyto nazwy istniejącego schematu.

Ograniczenia:

W zdalnych źródłach danych nie można tworzyć zmaterializowanych tabel zapytań.

W definicji tabeli można określić podstawowe informacje o kolumnach, jednak nie można określić opcji tabel ani opcji kolumn. Na przykład nie są obsługiwane opcje LOB (LOGGED i COMPACT).

Nie można określić komentarza dla kolumny.

Nie można wygenerować zawartości kolumny.

Można zdefiniować klucz podstawowy, jednak nie można określić klucza obcego ani ograniczeń sprawdzających. Kolumny użyte w charakterze klucza podstawowego muszą mieć przypisany atrybut NOT NULL i nie mogą zawierać danych typu LOB.

Nie jest obsługiwana klauzula DEFAULT w instrukcji CREATE TABLE.

Procedura:

Zdalną tabelę można utworzyć przy użyciu przezroczystego kodu DDL z Centrum sterowania lub z wiersza komend.

Aby utworzyć zdalne tabele z Centrum sterowania DB2, należy użyć Kreatora tworzenia tabeli zdalnej.

1. Rozwiń folder **Obiekty stowarzyszonej bazy danych**.
2. Rozwiń obiekty opakowań i definicji serwera dla źródła danych, dla którego chcesz utworzyć zdalną tabelę.
3. Kliknij prawym przyciskiem myszy folder **Tabele zdalne**, a następnie polecenie **Utwórz**. Zostanie uruchomiony Kreator tworzenia tabeli zdalnej.
4. Wykonaj instrukcje kreatora.

Aby wykonać to zadanie z wiersza komend, należy użyć instrukcji CREATE TABLE z odpowiednimi parametrami.

Zdalne źródło danych musi obsługiwać typy danych i opcje klucza podstawowego użyte w instrukcji CREATE TABLE. Załóżmy na przykład, że zdalne źródło danych nie umożliwi określenia kluczy podstawowych. Zależnie od sposobu, w jaki źródło danych odpowiada na nieobsługiwane żądania, może nastąpić wygenerowanie błędu lub zignorowanie żądania.

W klauzuli OPTIONS należy określić serwer zdalny. Za pomocą klauzuli OPTIONS można zastąpić zdalną nazwę lub zdalny schemat tworzonej tabeli.

Założmy, że tworzona jest tabela EMPLOYEE na serwerze Oracle. W specyfikacji poszczególnych kolumn w instrukcji CREATE TABLE należy używać typów danych DB2. Instrukcja tworząca tabelę za pomocą procesora wiersza komend będzie miała następującą postać:

```
CREATE TABLE EMPLOYEE
( EMP_NO      CHAR(6) NOT NULL,
  FIRSTNAME   VARCHAR(12) NOT NULL,
  MIDINT      CHAR(1) NOT NULL,
  LASTNAME    VARCHAR(15) NOT NULL,
  HIREDATE    DATE,
  JOB         CHAR(8),
  SALARY      DECIMAL(9,2),
  PRIMARY KEY (EMP_NO) )
OPTIONS (REMOTE_SERVER 'ORASERVER',
        REMOTE_SCHEMA 'J15USER1', REMOTE_TABNAME 'EMPLOY' )
```

EMPLOYEE

Lokalna nazwa tabeli. Ta sama nazwa służy też jako pseudonim skojarzony z tabelą.

REMOTE_SERVER 'ORASERVER'

Nazwa nadana serwerowi w instrukcji CREATE SERVER. W tej nazwie jest rozróżniana wielkość liter.

REMOTE_SCHEMA 'J15USER1'

Nazwa zdalnego schematu. Mimo, że parametr ten jest opcjonalny, jego użycie jest zalecane. W razie braku tej opcji jako nazwa zdalnego schematu zostaje domyślnie przyjęta wartość lokalnego identyfikatora AUTHID pisana wielkimi literami. W tej nazwie jest rozróżniana wielkość liter.

REMOTE_TABNAME 'EMPLOY'

Nazwa tabeli zdalnej. Parametr ten jest opcjonalny. W razie braku tej opcji jako nazwa zdalnej tabeli zostaje domyślnie przyjęta lokalna nazwa tabeli. Nazwa ta musi być poprawną nazwą tabeli według kryteriów zdalnego źródła danych i nie może wskazywać tabeli już istniejącej. W tej nazwie jest rozróżniana wielkość liter.

Podczas tworzenia zdalnej tabeli za pośrednictwem programu DB2 Information Integrator przy użyciu przezroczystych instrukcji DDL wykonywanych jest kilka innych działań:

- Automatycznie tworzony jest pseudonim odpowiadający zdalnej tabeli. Pseudonim ma taką samą nazwę, jak tabela lokalna. Tabela zdalna ma taką samą nazwę, jak tabela lokalna, chyba że określi się inną nazwę przy użyciu opcji REMOTE_TABNAME. Pseudonim zdalnej tabeli może być używany na takich samych zasadach, jak inne pseudonimy. Ponadto taką zdalną tabelę można modyfikować i usuwać (operacje ALTER i DROP), w przeciwieństwie do pseudonimów utworzonych za pomocą instrukcji CREATE NICKNAME.
- W widoku katalogu SYSCAT.TABOPTIONS dodawany jest wiersz z nazwą opcji TRANSPARENT i wartością '.

W powyższym przykładzie program DB2 Information Integrator korzysta ze zwrotnego odwzorowania typów danych w celu przypisania typów danych DB2 do typów danych Oracle. Na zdalnym serwerze Oracle tabela EMPLOY jest tworzona przy użyciu typów danych Oracle. Poniższa tabela przedstawia odwzorowania z typów danych DB2 na typy danych Oracle właściwe dla kolumn użytych w przykładzie.

Tabela 6. Przykład zwrotnych odwzorowań typów danych z programu DB2 Information Integrator do Oracle

Kolumna	Typ danych DB2 użyty w instrukcji CREATE TABLE	Typ danych Oracle używany w tabeli zdalnej
EMP_NO	CHAR(6) NOT NULL	CHAR(6) NOT NULL

Tabela 6. Przykład zwrotnych odwzorowań typów danych z programu DB2 Information Integrator do Oracle (kontynuacja)

Kolumna	Typ danych DB2 użyty w instrukcji CREATE TABLE	Typ danych Oracle używany w tabeli zdalnej
FIRST_NAME	VARCHAR(12) NOT NULL	VARCHAR2(12) NOT NULL
MID_INT	CHAR(1) NOT NULL	CHAR(1) NOT NULL
LAST_NAME	VARCHAR(15) NOT NULL	VARCHAR2(15) NOT NULL
HIRE_DATE	DATE	DATE
JOB	CHAR(8)	CHAR(8)
SALARY	DECIMAL(9,2)	NUMBER(9,2)

Na końcu instrukcji CREATE TABLE dopuszczalne jest wystąpienie opcji SQL_SUFFIX. Opcja ta służy przede wszystkim do określenia klauzuli IN TABLESPACE podczas tworzenia zdalnych tabel w źródłach danych z rodziny DB2.

Pojęcia pokrewne:

- “Co to jest przezroczysty kod DDL?” na stronie 81

Zadania pokrewne:

- “Modyfikowanie tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 86
- “Usuwanie tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 88

Informacje pokrewne:

- “CREATE TABLE statement” w podręczniku *SQL Reference, Volume 2*
- Rozdział 28, “Domyślne zwrotne odwzorowania typów”, na stronie 285

Modyfikowanie tabel zdalnych za pomocą przezroczystego kodu DDL

Instrukcje przezroczystego kodu DDL umożliwiają zmodyfikowanie w zdalnym źródle danych tabel, które zostały utworzone za pośrednictwem programu DB2 Information Integrator. Nie można modyfikować tabel, które zostały utworzone w zdalnym źródle danych za pomocą jego rodzimych mechanizmów.

Do modyfikowania tabel utworzonych przez program DB2 Information Integrator za pomocą instrukcji przezroczystego kodu DDL służy instrukcja ALTER TABLE. Za pomocą instrukcji ALTER TABLE można wykonać następujące operacje:

- dodawać nowe kolumny,
- dodać klucz podstawowy tabeli.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika skojarzony z instrukcjami przezroczystego kodu DDL musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie CREATETAB względem bazy danych i uprawnienie USE względem obszaru tabel, jak również jedno z uprawnień:
 - IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna bądź niejawną nazwa schematu tabeli;

- CREATEIN względem schematu, jeśli w instrukcji tworzenia tabeli użyto nazwy istniejącego schematu.

Aby wydawanie instrukcji przezroczystego kodu DDL było możliwe, niezbędne jest posiadanie przez dany identyfikator autoryzowanego użytkownika stosownych uprawnień względem pseudonimu (aby stowarzyszona baza danych przyjęła żądanie) oraz istnienie odpowiednich uprawnień na serwerze zdalnego źródła danych (aby żądanie zostało przyjęte przez źródło).

Ograniczenia:

Nie można modyfikować tabel, które zostały utworzone w zdalnym źródle danych za pomocą jego własnych mechanizmów.

Nie jest dopuszczalne usuwanie ani modyfikowanie istniejącego klucza podstawowego w tabeli zdalnej.

Modyfikowanie tabeli zdalnej powoduje unieważnienie wszystkich pakietów zależnych od pseudonimu skojarzonego z tą tabelą.

Zdalne źródło danych musi obsługiwać zmiany postulowane w instrukcji ALTER TABLE. Załóżmy na przykład, że zdalne źródło danych nie umożliwia określenia kluczy podstawowych. Zależnie od sposobu, w jaki źródło danych odpowiada na nieobsługiwane żądania, może nastąpić wygenerowanie błędu lub zignorowanie żądania.

Nie można określić komentarza dla kolumny.

Nie można wygenerować zawartości kolumny.

Można zdefiniować klucz podstawowy, jednak nie można określić klucza obcego ani ograniczeń sprawdzających. Kolumny użyte w charakterze klucza podstawowego muszą mieć przypisany atrybut NOT NULL i nie mogą zawierać danych typu LOB.

Nie można modyfikować parametrów istniejących kolumn, takich jak typ lub długość danych.

Nie jest obsługiwana klauzula DEFAULT w instrukcji ALTER TABLE.

Procedura:

Aby usunąć zdalną tabelę przy użyciu instrukcji przezroczystego kodu DDL, można posłużyć się Centrum sterowania DB2 lub instrukcją ALTER TABLE. Centrum sterowania DB2 eliminuje ryzyko użycia nieobsługiwanego parametru lub opcji.

Instrukcji ALTER TABLE nie należy używać do dodawania lub modyfikowania opcji kolumn. Do tego celu służy instrukcja ALTER NICKNAME.

Założmy, że musimy dodać klucz podstawowy do zdalnej tabeli EMPLOYEE utworzonej za pomocą instrukcji DDL. Instrukcja przezroczystego kodu wydana za pomocą procesora wiersza komend będzie miała następującą postać:

```
ALTER TABLE EMPLOYEE  
ADD PRIMARY KEY (EMP_NO, WORK_DEPT)
```

Kolumny wchodzące w skład klucza podstawowego muszą mieć przypisany atrybut NOT NULL i nie mogą zawierać danych typu LOB.

Załóżmy, że do zdalnej tabeli SPALTEN utworzonej za pomocą instrukcji przezroczystego kodu DDL trzeba dodać kolumny ORDER_DATE i SHIP_DATE. Instrukcja wydana za pomocą procesora wiersza komend będzie miała następującą postać:

```
ALTER TABLE SPALTEN
  ADD COLUMN ORDER_DATE DATE
  ADD COLUMN SHIP_DATE DATE
```

Pojęcia pokrewne:

- “Co to jest przezroczysty kod DDL?” na stronie 81

Zadania pokrewne:

- “Tworzenie nowych tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 83
- “Usuwanie tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 88

Informacje pokrewne:

- “ALTER TABLE statement” w podręczniku *SQL Reference, Volume 2*

Usuwanie tabel zdalnych za pomocą przezroczystego kodu DDL

Przezroczysty kod DDL umożliwia usunięcie w zdalnym źródle danych tabel, które zostały utworzone za pośrednictwem programu DB2 Information Integrator. Nie można usuwać tabel, które zostały utworzone w zdalnym źródle danych za pomocą jego rodzimych mechanizmów.

Wymagania wstępne:

Identyfikator autoryzowanego użytkownika skojarzony z instrukcjami przezroczystego kodu DDL musi mieć przyznane przynajmniej jedno z poniższych uprawnień:

- uprawnienie SYSADM lub DBADM
- uprawnienie CREATETAB względem bazy danych i uprawnienie USE względem obszaru tabel, jak również jedno z uprawnień:
 - IMPLICIT_SCHEMA względem bazy danych, jeśli nie istnieje jawna bądź niejawną nazwa schematu tabeli;
 - CREATEIN względem schematu, jeśli w instrukcji tworzenia tabeli użyto nazwy istniejącego schematu.

Aby wydawanie instrukcji przezroczystego kodu DDL było możliwe, niezbędne jest posiadanie przez dany identyfikator autoryzowanego użytkownika stosownych uprawnień względem pseudonimu (aby stowarzyszona baza danych przyjęła żądanie) oraz istnienie odpowiednich uprawnień na serwerze zdalnego źródła danych (aby żądanie zostało przyjęte przez źródło).

Ograniczenia:

Nie można usuwać tabel, które zostały utworzone w zdalnym źródle danych za pomocą jego własnych mechanizmów.

Procedura:

Aby usunąć zdalną tabelę, którą utworzono za pośrednictwem programu DB2 Information Integrator przy użyciu instrukcji przezroczystego kodu DDL, można posłużyć się Centrum sterowania DB2 lub instrukcją DROP. Usunięcie pseudonimu odpowiadającego zdalnej tabeli utworzonej za pomocą instrukcji przezroczystego kodu DDL skutkuje niczym więcej, jak

tylko usunięciem lokalnego pseudonimu tabeli. Instrukcja DROP NICKNAME nie powoduje usunięcia zdalnej tabeli. W celu usunięcia tabeli w źródle danych, należy użyć instrukcji DROP TABLE.

Usunięcie tabeli zdalnej powoduje najpierw usunięcie tabeli w źródle danych, a następnie usunięcie pseudonimu odpowiadającego tej tabeli w stowarzyszonej bazie danych. Usunięcie pseudonimu powoduje unieważnienie wszystkich bazujących na nim pakietów.

Instrukcja usuwająca tabelę SPALTEN miałaby następującą postać:

```
DROP TABLE SPALTEN
```

gdzie *SPALTEN* jest lokalną nazwą zdalnej tabeli.

Pojęcia pokrewne:

- “Co to jest przezroczysty kod DDL?” na stronie 81

Zadania pokrewne:

- “Tworzenie nowych tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 83
- “Modyfikowanie tabel zdalnych za pomocą przezroczystego kodu DDL” na stronie 86

Informacje pokrewne:

- “DROP statement” w podręczniku *SQL Reference, Volume 2*

Rozdział 7. Obsługa transakcji w systemach stowarzyszonych

Obsługa transakcji w systemach stowarzyszonych bazuje na funkcjach obsługi transakcji rozproszonych w programie DB2. W tym rozdziale zakłada się znajomość koncepcji dotyczących przetwarzania rozproszonych transakcji programu DB2, które opisano w następujących podręcznikach programu DB2:

- *DB2 Administration Guide: Planning (SC09-4822-00)*
- *DB2 Application Development Guide: Programming Client Applications (SC09-4826-00)*
- *DB2 Application Development Guide: Programming Server Applications (SC09-4827-00)*

Ten rozdział zawiera następujące sekcje:

- “Obsługa transakcji w systemie stowarzyszonym”
- “Aktualizacja w systemie stowarzyszonym” na stronie 92

Obsługa transakcji w systemie stowarzyszonym

Podstawą przetwarzania transakcji w systemie stowarzyszonym są funkcje rozproszonego przetwarzania transakcyjnego baz danych programu DB2®. Do zrozumienia zasad przetwarzania transakcji w systemie stowarzyszonym konieczna jest znajomość następujących podstawowych pojęć z zakresu rozproszonego przetwarzania transakcyjnego:

- jednostka pracy (UOW)
- zdalna jednostka pracy (RUOW)
- rozproszona jednostka pracy (DUOW)
- aktualizacja na wielu serwerach
- menedżer transakcji (TM)
- menedżer zasobów (RM)
- połączenie typu 1
- połączenie typu 2
- zatwierdzanie jednofazowe
- zatwierdzanie dwufazowe

Pojęcia te mają takie same znaczenie we wszystkich systemach DB2, zarówno stowarzyszonych, jak i niestowarzyszonych. W systemie stowarzyszonym zmienia się jednak zasięg tych pojęć.

Na przykład jednostka pracy rozpoczyna się niejawnie w momencie odczytu lub zapisu dowolnych danych w bazie danych. W systemie stowarzyszonym baza ta może być stowarzyszoną bazą danych lub bazą danych źródła danych. W wypadku rozproszonej jednostki pracy w systemie stowarzyszonym możliwy jest dostęp zarówno do stowarzyszonej bazy danych, jak i bazy danych źródła danych.

Aplikacja musi zakończyć jednostkę pracy za pomocą instrukcji COMMIT albo ROLLBACK, niezależnie od tego, z ilu baz danych korzysta. Instrukcja COMMIT utrwala wszystkie zmiany wprowadzone przez jednostkę pracy. Instrukcja ROLLBACK wycofuje te zmiany z bazy danych. Zmiany wprowadzone przez jednostkę pracy stają się widoczne dla innych aplikacji dopiero po pomyślnym zatwierdzeniu.

Zalecenie: W swoich aplikacjach należy zawsze w sposób jawny zatwierdzać lub wycofywać jednostki pracy.

Gdy jednostka pracy aktualizuje tylko jedno źródło danych, instrukcję COMMIT dla tej jednostki pracy można wysłać w jednej operacji. Operacja taka nazywa się *zatwierdzeniem jednofazowym* lub aktualizacją na jednym serwerze. Określenie *ośrodek* odpowiada definicji serwera w systemie stowarzyszonym. Serwer stowarzyszony jest ośrodkiem lokalnym dla operacji aktualizowania w systemie stowarzyszonym. Każde zdalne źródło danych to ośrodek zdalny dla operacji aktualizowania w systemie stowarzyszonym.

W operacji zatwierdzania jednofazowego żadne inne źródła danych w tej samej jednostce pracy nie muszą sprawdzać swojej gotowości do zatwierdzenia danych.

Przykładowe operacje zatwierdzania jednofazowego to:

- jednostka pracy, niebędąca jednostką rozproszoną;
- rozproszona jednostka pracy, w której dane mogą być odczytywane z wielu źródeł danych, ale która aktualizuje tylko jedno źródło danych;

W rozproszonej jednostce pracy, która aktualizuje wiele baz danych na wielu serwerach, dane muszą być spójne. Mechanizm aktualizacji na wielu serwerach, czyli protokół zatwierdzania dwufazowego, jest powszechnie używany do zapewnienia spójności danych w wielu bazach danych w ramach rozproszonej jednostki pracy. Obecnie systemy stowarzyszone nie obsługują protokołu zatwierdzania dwufazowego.

Pojęcia pokrewne:

- “Jednostki pracy” w podręczniku *Administration Guide: Planning*
- “DB2 transaction manager” w podręczniku *Administration Guide: Planning*
- “Two-phase commit” w podręczniku *Administration Guide: Planning*
- “Resource manager setup” w podręczniku *Administration Guide: Planning*
- “Multisite Update” w podręczniku *Application Development Guide: Programming Client Applications*
- “Aktualizacja w systemie stowarzyszonym” na stronie 92

Zadania pokrewne:

- “Updating a database from a host or iSeries client” w podręczniku *Administration Guide: Planning*
- “Updating a single database in a transaction” w podręczniku *Administration Guide: Planning*

Aktualizacja w systemie stowarzyszonym

W systemie stowarzyszonym aktualizacja nie jest zwykłą transakcją, która zawiera instrukcję INSERT, UPDATE lub DELETE. Istnieją pewne działania, w tym niektóre transakcje zawierające inne instrukcje, które w wypadku serwera stowarzyszonego są również uznawane za aktualizacje.

Aktualizacje mogą być wykonywane lokalnie lub zdalnie.

Aktualizacje lokalne są to aktualizacje obiektów w stowarzyszonej bazie danych. Obiektami lokalnymi są tabele i widoki w stowarzyszonej bazie danych. Dla tych obiektów nie tworzy się pseudonimów, lecz używa się w instrukcjach ich rzeczywistych nazw.

Aktualizacje zdalne są to aktualizacje obiektów w zdalnym źródle danych. Do zdalnych źródeł danych należą:

- inna baza danych lub instancja programu DB2[®] dla systemów Linux, UNIX[®] i Windows[®] na serwerze stowarzyszonym,
- inna baza danych lub instancja programu DB2 dla systemów Linux, UNIX i Windows na innym serwerze,
- źródła danych programów innych niż DB2 dla systemów Linux, UNIX i Windows, takich jak DB2 for iSeries[™], Informix[®], Oracle i Teradata.

Ważne jest zrozumienie, na czym polega transakcja aktualizacji w aspekcie serwera stowarzyszonego oraz poznanie typów aktualizacji dozwolonych w systemie stowarzyszonym. Wyróżnia się cztery typy działań, które serwer stowarzyszony uznaje za transakcje aktualizacji. W poniższej tabeli przedstawiono aktualizacje, które można wykonać w systemie stowarzyszonym.

Tabela 7. Rodzaje aktualizacji i serwery, na których są one wykonywane

Rodzaj działania	Serwer lokalny	Serwer zdalny	Objaśnienie
Lokalna aktualizacja (DDL i DML)	T	N	Aktualizacja obiektu w stowarzyszonej bazie danych.
Zdalna aktualizacja (pseudonim)	N	T	Aktualizacja obiektu w zdalnym źródle danych, dla którego utworzono pseudonim.
Dynamiczny SQL w sesjach tranzytowych	N	T	Aktualizacja obiektu zdalnego źródła danych. W sesji tranzytowej nie można aktualizować obiektów lokalnych.
Przezroczysty kod DDL	T	T	Para transakcji tworzących, zmieniających lub usuwających tabele i odpowiadające im pseudonimy w stowarzyszonej bazie danych. Na przykład para transakcji, które tworzą zdalną tabelę w źródle danych i pseudonim na serwerze stowarzyszonym.

Transakcja aktualizacji w sesji tranzytowej

Serwer stowarzyszony uznaje za aktualizacje wszystkie instrukcje dynamicznego SQL wysyłane w sesji tranzytowej. Takie zachowanie zapewnia integralność danych. Jeśli instrukcja dynamicznego SQL wysłana w sesji tranzytowej zostanie wykonana pomyślnie, transakcja jest rejestrowana jako aktualizacja. Typ instrukcji SQL nie ma w tym wypadku znaczenia, może to być na przykład instrukcja SELECT.

Obsługa transakcji realizowanych przy użyciu przezroczystego kodu DDL

Przezroczysty DDL tworzy tabelę w zdalnym źródle danych i pseudonim dla tej zdalnej tabeli w lokalnej stowarzyszonej bazie danych. Instrukcje przezroczystego kodu DDL tworzą tabele w zdalnym źródle danych i pseudonimy dla tych zdalnych tabel w lokalnej stowarzyszonej bazie danych. Serwer stowarzyszony traktuje transakcje przezroczystego kodu DDL jak aktualizacje.

Przed transakcjami realizowanymi przy użyciu przezroczystego kodu DDL i po nich konieczne jest użycie instrukcji COMMIT lub ROLLBACK.

Ponieważ przezroczysty kod DDL tworzy jednocześnie obiekty lokalne i zdalne, każda użyta instrukcja przezroczystego kodu DDL musi być jedyną aktualizacją w ramach transakcji zatwierdzania jednofazowego. Jeśli przed transakcją przezroczystego kodu DDL wystąpi inna aktualizacja, bezpośrednio po niej musi być użyta instrukcja COMMIT lub ROLLBACK. Podobnie instrukcja COMMIT lub ROLLBACK musi być użyta zaraz po transakcji realizowanej przy użyciu przezroczystego kodu DDL, aby możliwe było przeprowadzenie innej aktualizacji.

Źródła danych automatycznie zatwierdzające instrukcje DDL

Niektóre źródła danych automatycznie zatwierdzają lokalnie bieżącą transakcję DDL, która zakończyła się pomyślnie. W takich źródłach danych niemożliwe jest wycofanie zdalnej tabeli utworzonej przy użyciu przezroczystego kodu DDL lub w sesji tranzytowej. Konieczne jest ręczne usunięcie zdalnej tabeli.

Jedynе stowarzyszone źródło danych, które automatycznie zatwierdza instrukcje DDL, to źródło danych Oracle.

Funkcje zdefiniowane przez użytkownika przekazywane do przetwarzania w źródle danych

Serwer stowarzyszony traktuje funkcje zdefiniowane przez użytkownika, które są przekazywane do źródła danych, jako instrukcje odczytu. Jeśli zdalna funkcja zdefiniowana przez użytkownika wykona aktualizację źródła danych, serwer stowarzyszony nie zostanie o tym poinformowany. Ponieważ serwer stowarzyszony nie traktuje takich funkcji jak instrukcji aktualizacji, niemożliwe jest wycofanie tych instrukcji.

Ważne: Nie można zagwarantować integralności danych, gdy funkcja zdefiniowana przez użytkownika jest przekazywana do źródła danych w celu przeprowadzenia aktualizacji.

Pojęcia pokrewne:

- “Bezpośrednie kierowanie zapytań do źródeł danych w sesji tranzytowej” na stronie 213
- “Obsługa transakcji w systemie stowarzyszonym” na stronie 91

Rozdział 8. Operacje INSERT, UPDATE i DELETE

W tym rozdziale opisano, w jaki sposób uzyskiwać dostęp do danych w źródłach danych, i w jaki sposób aktualizować te dane.

Ten rozdział zawiera:

- “Uprawnienia wymagane do użycia instrukcji INSERT, UPDATE i DELETE”
- “Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym” na stronie 96
- “Spójność referencyjna w systemie stowarzyszonym” na stronie 97
- “Instrukcje INSERT, UPDATE i DELETE a obiekty LOB” na stronie 97
- “Zachowywanie niepodzielności instrukcji w systemie stowarzyszonym” na stronie 98
- “Praca z pseudonimami” na stronie 100
- “Selekcja danych w systemie stowarzyszonym” na stronie 110
- “Wstawianie danych do obiektów źródeł danych” na stronie 113
- “Aktualizowanie danych w obiektach źródła danych” na stronie 114
- “Usuwanie danych z obiektów źródeł danych” na stronie 115

Uprawnienia wymagane do użycia instrukcji INSERT, UPDATE i DELETE

Uprawnienia konieczne do wydawania instrukcji INSERT, UPDATE i DELETE względem pseudonimów są zbliżone do uprawnień wymaganych w stosunku do tabel. Dodatkowo niezbędne jest dysponowanie odpowiednimi uprawnieniami w źródle danych, zezwalającymi na wykonywanie operacji wstawiania, aktualizacji lub usuwania na skojarzonym z pseudonimem obiekcie.

Uprawnienia SELECT, INSERT, UPDATE i DELETE względem pseudonimu można nadawać i odbierać.

Jednak nadawanie lub odbieranie uprawnień względem pseudonimu nie powoduje nadania lub odebrania odpowiednich uprawnień w źródle danych. W zdalnym źródle danych czynności nadawania i odbierania uprawnień muszą dotyczyć identyfikatora REMOTE_AUTHID określonego w odwzorowaniu użytkowników na serwerze stowarzyszonym.

Aby stowarzyszona baza danych przyjęła żądanie, identyfikator autoryzowanego użytkownika, z którym skojarzona jest instrukcja, musi mieć przyznane niezbędne uprawnienia względem pseudonimu. Identyfikator użytkownika w źródle danych, który jest odwzorowany na bieżący identyfikator autoryzowanego użytkownika, musi mieć wymagane uprawnienia względem tabeli związanej z pseudonimem (aby żądanie zostało przyjęte przez źródło danych).

Uprawnienia względem pseudonimów używanych w zapytaniu są sprawdzane podczas wprowadzania zapytania do stowarzyszonej bazy danych. Uprawnienia względem obiektu źródła danych, z którym skojarzony jest pseudonim, są egzekwowane dopiero w momencie faktycznej realizacji zapytania. Użytkownik pozbawiony uprawnień SELECT względem pseudonimu nie będzie mógł wybierać rekordów ze źródła danych, do którego pseudonim się odwołuje.

Analogicznie, uprawnienie UPDATE względem pseudonimu jest równoznaczne z uprawnieniem do modyfikacji reprezentowanego przez pseudonim obiektu w źródle danych. Pomyślny wynik kontroli uprawnień na serwerze stowarzyszonym nie oznacza, że również pomyślny będzie wynik sprawdzenia uprawnień w zdalnym źródle danych. Za pośrednictwem odwzorowania użytkowników identyfikator autoryzowanego użytkownika z serwera stowarzyszonego jest odwzorowywany na identyfikator zdefiniowany w źródle danych. Uprawnienia będą egzekwowane w źródle danych.

Zadania pokrewne:

- “Modyfikowanie odwzorowania użytkowników” na stronie 32
- “Instrukcje SQL używane w odniesieniu do pseudonimów” na stronie 101

Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym

W systemie stowarzyszonym aktualizacjami są nie tylko transakcje inicjowane operacjami INSERT, UPDATE lub DELETE. Również pewne inne działania są przez serwer stowarzyszony uznawane za aktualizacje. Dotyczy to w szczególności niektórych transakcji z instrukcjami SELECT. Patrz Aktualizacja w systemie stowarzyszonym.

Aktualizacje mogą być wykonywane lokalnie lub zdalnie. Aktualizacje lokalne są modyfikacjami obiektów stowarzyszonej bazy danych. Obiektami lokalnymi są tabele i widoki w stowarzyszonej bazie danych. Aktualizacje zdalne są to aktualizacje obiektów w zdalnym źródle danych.

Aktualizacje pseudonimów podlegają następującym ograniczeniom:

- Obiekt, którego źródło danych nie umożliwia aktualizacji, nie może być aktualizowany.
- Nie jest możliwe aktualizowanie obiektu przeznaczonego tylko do odczytu, takiego jak widok JOIN.
- Nie jest możliwe wykonywanie operacji wstawiania, aktualizacji ani usuwania na widokach stowarzyszonych utworzonych za pomocą instrukcji UNION ALL. Widoki stowarzyszone utworzone za pomocą instrukcji UNION ALL są przeznaczone tylko do odczytu.

Nieobsługiwane źródła danych

System stowarzyszony nie umożliwia wykonywania operacji wstawiania, aktualizacji ani usuwania w następujących źródłach danych:

- BLAST
- BioRS
- Documentum
- Entrez
- Excel
- Extended Search
- HMMER
- Pliki o strukturze tabeli
- Usługi WWW
- WebSphere® Business Integration
- XML

Pojęcia pokrewne:

- “Uprawnienia wymagane do użycia instrukcji INSERT, UPDATE i DELETE” na stronie 95
- “Spójność referencyjna w systemie stowarzyszonym” na stronie 97
- “Instrukcje INSERT, UPDATE i DELETE a obiekty LOB” na stronie 97

Spójność referencyjna w systemie stowarzyszonym

W systemie stowarzyszonym program DB2® UDB nie zapewnia kompensacji naruszeń spójności referencyjnej między źródłami danych. Program DB2 UDB nie ingeruje w działanie mechanizmów spójności referencyjnej w źródłach danych.

Tym niemniej ograniczenia spójności referencyjnej w źródle danych mogą mieć wpływ na aktualizacje pseudonimu. Załóżmy, że do stowarzyszonej bazy danych wpływa żądanie wstawienia danych do pseudonimu. Po przesłaniu żądania przez serwer stowarzyszony do źródła danych okazuje się, że operacja ta narusza zasady spójności referencyjnej w tym źródle danych. Program DB2 UDB odwzorowuje wynikowy błąd na błąd DB2 UDB.

Odpowiedzialność za utrzymanie spójności referencyjnej między źródłami danych spoczywa na aplikacjach.

Program DB2 Information Integrator obsługuje ograniczenia informacyjne, za pośrednictwem których można poinformować procesor zapytań o ograniczeniach spójności referencyjnej.

Pojęcia pokrewne:

- “Ograniczenia informacyjne pseudonimów” na stronie 185
- “Jak aplikacje klienckie współpracują ze źródłami danych” na stronie 201
- “Zachowywanie niepodzielności instrukcji w systemie stowarzyszonym” na stronie 98

Instrukcje INSERT, UPDATE i DELETE a obiekty LOB

Trzy typy danych LOB w programie DB2® UDB to: CLOB (duży obiekt znakowy), DBCLOB (duży obiekt złożony ze znaków dwubajtowych) oraz BLOB (duży obiekt binarny).

Dzięki stowarzyszeniu można wykonywać operacje odczytu obiektów LOB znajdujących się we wszystkich relacyjnych źródłach danych. Operacje zapisu obiektów LOB w źródłach danych Oracle (wersja 8 lub nowsza) są możliwe przy użyciu opakowania NET8.

Pojęcia pokrewne:

- “Stowarzyszona obsługa obiektów LOB” na stronie 208
- “Wskaźniki LOB” na stronie 210
- “Ograniczenia dotyczące obiektów LOB” na stronie 210

Informacje pokrewne:

- “Zmiany długich typów danych na typy varchar” na stronie 57

Zachowywanie niepodzielności instrukcji w systemie stowarzyszonym

Podczas operacji aktualizacji systemy stowarzyszone zawsze próbują utrzymać dane w stanie niepodzielności, aż do zakończenia wykonywania instrukcji języka DML. Kiedy dane są w stanie niepodzielności, można zagwarantować ich pomyślne przetworzenie lub zachowanie w stanie niezmienionym.

Gdy klient lub aplikacja wyśle instrukcję INSERT, UPDATE lub DELETE dla pseudonimu, serwer stowarzyszony wewnętrznie przetwarza tę instrukcję jako pojedynczą instrukcję DML lub jako serię wielu instrukcji DML. Jeśli serwer stowarzyszony musi wysłać wiele instrukcji DML do przetworzenia w docelowym źródle danych, niepodzielność danych jest potencjalnie zagrożona. Aby temu zapobiec, systemy stowarzyszone korzystają z wywołań API punktu zapisu źródła danych w celu monitorowania serii wielu instrukcji DML.

Niektóre źródła danych lub niektóre wersje źródeł danych nie udostępniają na zewnątrz funkcji API punktu zapisu, które mogłyby być użyte przez system stowarzyszony. W takich sytuacjach stowarzyszone instrukcje INSERT, UPDATE lub DELETE są wykonywane bez ochrony interfejsów API punktu zapisu.

Gdy podczas stowarzyszonych transakcji wstawiania, aktualizacji lub usuwania wystąpi błąd, jego skutkiem może być częściowa aktualizacja w źródłach danych. Aby skorygować problemy niespójności, system stowarzyszony automatycznie wewnętrznie wycofuje transakcję zanim zwróci kod błędu SQL do aplikacji.

Poniższe źródła danych nie udostępniają na zewnątrz interfejsów API punktu zapisu, które mogłyby być używane przez serwer stowarzyszony:

- DB2[®] dla systemów Linux, UNIX[®] i Windows[®], wersja 6 (i starsze)
- DB2 for iSeries[™]
- DB2 for VM and VSE
- DB2 for z/OS[™] and OS/390[®], wersja 5 (i starsze)
- Informix[®]
- Microsoft[®] SQL Server
- ODBC
- Teradata

Jeśli cała transakcja wstawiania, aktualizacji lub usuwania jest przesyłana do przetworzenia w źródle danych, serwer stowarzyszony zakłada, że źródło danych zachowa niepodzielność instrukcji w wypadku błędu.

Gdy tylko część transakcji wstawiania, aktualizacji lub usuwania jest przesyłana do przetworzenia w źródle danych, to w wypadku błędu zostanie wycofana cała transakcja.

Działaniami tymi steruje opcja serwera IUD_APP_SVPT_ENFORCE. Wartość domyślna tej opcji to 'Y', co oznacza wycofanie transakcji w wypadku błędu.

Użycie instrukcji SET SERVER OPTION do ustawienia opcji serwera IUD_APP_SVPT_ENFORCE na czas trwania połączenia nie odniesie żadnego skutku w przypadku statycznych instrukcji SQL. Ustawienie tej opcji serwera za pomocą instrukcji SET SERVER OPTION będzie miało wpływ tylko na dynamiczne instrukcje SQL.

Scenariusz działania dla opcji serwera IUD_APP_SVPT_ENFORCE:

Założmy, że dla tabeli Informix o nazwie UT utworzono pseudonim INFMX_UT. Tabela UT zawiera cztery kolumny typu integer: i1, i2, i3 oraz i4. Kolumna i1 to kolumna indeksu unikalnego.

Tabela UT jest pusta. Względem pseudonimu INFX_UT zostaje użyta instrukcja INSERT w celu wstawienia w pierwszym wierszu tabeli wartości 1, 22, 34 i 40. Instrukcja ta zostaje wykonana pomyślnie.

Następnie względem pseudonimu INFX_UT zostaje użytych szereg instrukcji INSERT, w celu wstawienia trzech wierszy danych:

- Wiersz 2: 2, 37, 34, 55
- Wiersz 3: 3, 42, 59, 40
- Wiersz 4: 1, 55, 62, 75

Ponieważ dane, które mają być wstawione w ostatnim wierszu, nie spełniają wymogu unikalności indeksu w kolumnie i1, serwer Informix zwraca komunikat o błędzie do systemu stowarzyszonego. Z kolei system stowarzyszony zwraca do aplikacji błąd SQL o kodzie SQL0803N. Komunikat o błędzie SQL0803N opisuje naruszenie unikalności indeksów.

Poniżej przedstawiono wiersze tabeli UT po operacjach wstawiania:

Tabela 8. Przykład tabeli UT w źródle danych Informix

	Kolumna (indeks unikalny)	Kolumna	Kolumna	Kolumna
Wiersz	i1	i2	i3	i4
Wiersz 1	1	22	34	40
Wiersz 2	2	37	34	55
Wiersz 3	3	42	59	40

Domyślne działanie opcji serwera IUD_APP_SVPT_ENFORCE:

Wartością domyślną opcji serwera IUD_APP_SVPT_ENFORCE jest 'Y'. Wartość ta oznacza wewnętrzne wycofanie całej transakcji. Chociaż dwa pierwsze wiersze danych zostały pomyślnie wstawione, zostaną one wycofane, ponieważ wycofywana jest cała transakcja.

Aby zmienić domyślną wartość opcji serwera IUD_APP_SVPT_ENFORCE, należy użyć instrukcji ALTER SERVER. Zmiana ta dotyczy wszystkich obiektów źródła danych, do których uzyskiwany jest dostęp za pośrednictwem określonego serwera.

Alternatywne zachowanie opcji serwera IUD_APP_SVPT_ENFORCE:

Kiedy opcja serwera IUD_APP_SVPT_ENFORCE ma wartość 'N', transakcja nie jest wycofywana. Dane wstawione w drugim i trzecim wierszu pozostaną w tabeli. Za obsługę błędu odpowiada aplikacja.

Pojęcia pokrewne:

- “Spójność referencyjna w systemie stowarzyszonej” na stronie 97

Informacje pokrewne:

- Rozdział 21, “Opcje serwera w systemie stowarzyszonej”, na stronie 225

Praca z pseudonimami

Chcąc wybrać dane ze źródła danych lub zmienić te dane, tworzy się zapytania dla pseudonimów zawierające instrukcje SELECT, INSERT, UPDATE i DELETE. Zapytania te tworzy się w języku SQL programu DB2 i wprowadza do stowarzyszonej bazy danych. Za pomocą jednej instrukcji SQL można połączyć dane z tabeli lokalnej i zdalnego źródła danych, a następnie korzystać z nich tak, jak gdyby wszystkie te dane były przechowywane lokalnie. Można na przykład łączyć dane znajdujące się w:

- lokalnej tabeli stowarzyszonej bazy danych DB2 for Windows, tabeli bazy danych Oracle i widoku bazy danych Sybase;
- tabeli bazy danych DB2 UDB for z/OS na jednym serwerze, tabeli bazy danych DB2 UDB for z/OS na innym serwerze i arkusza kalkulacyjnym programu Excel.

Przetwarzając instrukcje SQL tak, jak gdyby źródła danych były zwykłymi tabelami relacyjnymi lub widokami w stowarzyszonej bazie danych, system stowarzyszony może łączyć dane systemów relacyjnych z danymi w formatach nierelacyjnych.

Obiektami lokalnymi są tabele i widoki w stowarzyszonej bazie danych. Dla tych obiektów nie należy tworzyć pseudonimów, tylko używać w instrukcjach ich rzeczywistych nazw.

Obiekty zdalne to obiekty, które znajdują się poza stowarzyszoną bazą danych. Dla tych obiektów konieczne jest utworzenie pseudonimów. Są to na przykład:

- tabele i widoki w innej bazie danych lub instancji DB2 dla systemów Linux, UNIX i Windows w systemie stowarzyszonym,
- tabele i widoki w innej bazie danych lub instancji DB2 dla systemów Linux, UNIX i Windows w innym systemie,
- obiekty znajdujące się w źródłach danych innych niż bazy danych DB2 dla systemów Linux, UNIX i Windows, takich jak: Oracle, Sybase, Documentum i źródła ODBC.

Procedura:

Do wykorzystania możliwości systemu stowarzyszonego potrzebna jest wiedza na temat:

- instrukcji SQL, których można używać w odniesieniu do pseudonimu;
- metod dostępu do nowych obiektów źródeł danych;
- sytuacji, w których uzyskuje się bezpośredni dostęp do danych w sesji tranzytowej;
- zalet korzystania z widoków w systemie stowarzyszonym w celu uzyskania dostępu do heterogenicznych źródeł danych.

Składnia WITH HOLD

| W odniesieniu do kursora zdefiniowanego dla pseudonimu można użyć składni WITH
| HOLD. Jeśli jednak źródło danych nie obsługuje tej składni, próba jej użycia (z instrukcją
| COMMIT) spowoduje wyświetlenie komunikatu o błędzie.

Wyzwalacze

| Pseudonim nie może być celem aktualizacji w wyzwalaczu. W treści wyzwalacza można
| zamieścić instrukcje SELECT odnoszące się do pseudonimów. Nie można natomiast
| zamieścić tam instrukcji INSERT, UPDATE lub DELETE dotyczących pseudonimów.

Pojęcia pokrewne:

- “Upewnienia wymagane do użycia instrukcji INSERT, UPDATE i DELETE” na stronie 95

- “Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym” na stronie 96

Zadania pokrewne:

- “Odwoływanie się do obiektów źródeł danych za pośrednictwem pseudonimów w instrukcjach SQL” na stronie 202
- “Instrukcje SQL używane w odniesieniu do pseudonimów” na stronie 101
- “Dostęp do nowych obiektów źródeł danych” na stronie 105
- “Dostęp do źródeł danych przy użyciu sesji tranzytowych” na stronie 107
- “Uzyskiwanie dostępu do danych heterogenicznych za pośrednictwem widoków stowarzyszonych” na stronie 108

Praca z pseudonimami - informacje szczegółowe

W tej sekcji opisano operacje wstawiania, aktualizowania i usuwania wykonywane na pseudonimach.

Instrukcje SQL używane w odniesieniu do pseudonimów

W systemie stowarzyszonym można w łatwy sposób uzyskać dostęp do danych, niezależnie od tego, gdzie się one rzeczywiście znajdują. W tym celu tworzy się pseudonimy dla wszystkich obiektów źródła danych (takich jak tabele i widoki).

Jeśli na przykład zostanie utworzony pseudonim DEPT reprezentujący zdalną tabelę EUROPE.PERSON.DEPT, w zapytaniu wybierającym informacje ze zdalnej tabeli będzie można użyć instrukcji SELECT * FROM DEPT. Użycie w zapytaniu pseudonimu eliminuje konieczność zapamiętywania informacji o rzeczywistym źródle danych. Dzięki pseudonimom użytkownicy podczas tworzenia zapytań nie muszą brać pod uwagę takich kwestii, jak:

- nazwa obiektu w źródle danych;
- serwer, na którym rezyduje źródło danych;
- typ źródła danych, w którym rezydują obiekty, na przykład Informix lub Oracle;
- język zapytań lub dialekt SQL używany w źródle danych;
- odwzorowania typów danych między źródłem danych i bazą danych DB2 Information Integrator;
- odwzorowania funkcji między źródłem danych a bazą danych DB2 Information Integrator.

Wszystkie informacje niezbędne serwerowi stowarzyszonemu do przetwarzania zapytań są dostarczane za pośrednictwem metadanych przechowywanych w katalogu stowarzyszonej bazy danych. Te metadane są zbierane ze źródeł danych podczas definiowania serwera stowarzyszonego oraz stowarzyszonej bazy danych i konfigurowania w nich dostępu do źródeł danych.

Po skonfigurowaniu systemu stowarzyszonego pseudonimów używa się w zapytaniach kierowanych do źródeł danych oraz do dalszego rozszerzenia konfiguracji systemu stowarzyszonego.

W poniższej tabeli wymieniono instrukcje SQL, w których można korzystać z pseudonimów:

Tabela 9. Typowe instrukcje SQL obsługujące pseudonimy

Instrukcja SQL	Opis	Wymagane uprawnienia
ALTER NICKNAME	Modyfikuje istniejący pseudonim, zmieniając lokalną nazwę kolumny, lokalny typ danych, opcje kolumny stowarzyszonej lub ograniczenia informacyjne. Zmiany te nie mają wpływu na tabelę lub widok w źródle danych.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie ALTER lub CONTROL względem pseudonimu • uprawnienie ALTERIN względem schematu, o ile nazwa schematu pseudonimu istnieje • uprawnienia użytkownika definiującego pseudonim, zgodnie z zawartością kolumny DEFINER w widoku katalogu pseudonimu
ALTER TABLE	Zmienia zdalną tabelę utworzoną w programie DB2 Information Integrator przy użyciu przezroczystego kodu DDL. Nie można zmieniać tabel utworzonych w źródle danych w trybie rodzimym. Umożliwia zastosowanie ograniczeń informacyjnych w celu dodania do pseudonimu ograniczenia spójności referencyjnej.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie ALTER lub CONTROL względem pseudonimu • uprawnienie ALTERIN względem schematu, o ile nazwa schematu pseudonimu istnieje
COMMENT ON	Dodaje lub zamienia komentarze w katalogowych opisach różnych obiektów, w tym funkcji, odwzorowań funkcji, indeksów, pseudonimów, serwerów, opcji serwera, odwzorowań typów, opakowań.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie ALTER lub CONTROL względem obiektu • uprawnienie ALTERIN względem schematu • uprawnienia użytkownika definiującego pseudonim, zgodnie z zawartością kolumny DEFINER w widoku katalogu pseudonimu
CREATE ALIAS	Definiuje alias dla pseudonimu.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie IMPLICIT_SCHEMA względem bazy danych, o ile nie istnieje jawna lub niejawna nazwa schematu aliasu • uprawnienie CREATEIN względem schematu, o ile nazwa schematu aliasu odwołuje się do istniejącego schematu
CREATE INDEX (z klauzulą SPECIFICATION ONLY)	Tworzy specyfikację indeksu (metadane), która informuje optymalizator zapytań, że źródło danych ma indeks. Instrukcja ta nie tworzy rzeczywistego indeksu, a tylko jego specyfikację.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie CONTROL lub INDEX względem odpowiedniego obiektu źródła danych — <i>oraz jedno z następujących</i>: uprawnienie IMPLICIT_SCHEMA względem bazy danych lub uprawnienie CREATEIN względem schematu

Tabela 9. Typowe instrukcje SQL obsługujące pseudonimy (kontynuacja)

Instrukcja SQL	Opis	Wymagane uprawnienia
CREATE TABLE (z klauzulą OPTIONS)	Tworzy zdalną tabelę utworzoną w programie DB2 Information Integrator przy użyciu przezroczystego kodu DDL.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie CREATETAB względem bazy danych i uprawnienie USE względem obszaru tabel — <i>oraz jedno z następujących</i>: uprawnienie IMPLICIT_SCHEMA względem bazy danych lub uprawnienie CREATEIN względem schematu
CREATE TABLE (z klauzulą pełnej selekcji AS i z klauzulą DATA INITIALLY DEFERRED REFRESH)	Tworzy zmaterializowaną tabelę zapytania przy użyciu pełnej selekcji odwołującej się do pseudonimu.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie CREATETAB względem bazy danych i uprawnienie USE względem obszaru tabel — <i>oraz jedno z następujących</i>: uprawnienie IMPLICIT_SCHEMA względem bazy danych lub uprawnienie CREATEIN względem schematu • uprawnienie CONTROL względem tabeli lub widoku • uprawnienie SELECT względem tabeli lub widoku i uprawnienie ALTER, o ile określono opcję REFRESH DEFERRED lub REFRESH IMMEDIATE
CREATE VIEW	Tworzy widok odwołujący się do co najmniej jednego pseudonimu.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie CONTROL lub SELECT względem pseudonimu — <i>oraz jedno z następujących</i>: uprawnienie IMPLICIT_SCHEMA względem bazy danych lub uprawnienie CREATEIN względem schematu
DELETE	Usuwa wiersze z obiektu źródła danych (takiego jak tabela lub widok), dla którego zdefiniowano pseudonim.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie DELETE względem pseudonimu i uprawnienie DELETE względem odpowiedniego obiektu źródła danych • uprawnienie CONTROL względem odpowiedniego obiektu źródła danych
DROP	<p>Usuwa specyfikację indeksu obiektu, takiego jak pseudonim lub widok stowarzyszony. Zmiany te nie mają wpływu na tabelę, widok ani indeks w źródle danych.</p> <p>Usunięcie tabeli utworzonej przy użyciu przezroczystego kodu DDL powoduje usunięcie odpowiadającego jej pseudonimu.</p>	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie DROPIN względem schematu dla obiektu • uprawnienie CONTROL względem obiektu

Tabela 9. Typowe instrukcje SQL obsługujące pseudonimy (kontynuacja)

Instrukcja SQL	Opis	Wymagane uprawnienia
GRANT	Nadaje uprawnienia względem pseudonimów i widoków stowarzyszonych, takie jak ALTER, DELETE, INDEX, INSERT, SELECT, UPDATE. Uprawnienia względem obiektów źródła danych muszą być nadawane osobno.	<ul style="list-style-type: none"> • SYSADM lub DBADM • WITH GRANT OPTION dla każdego nadawanego uprawnienia • uprawnienie CONTROL względem obiektu
INSERT	Wstawia wiersze do obiektu źródła danych (takiego jak tabela lub widok), dla którego zdefiniowano pseudonim.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie INSERT względem pseudonimu i uprawnienie INSERT względem odpowiedniego obiektu źródła danych • uprawnienie CONTROL względem odpowiedniego obiektu źródła danych
LOCK TABLE	Powoduje zablokowanie zdalnego obiektu w źródle danych. Uniemożliwia współbieżnie działającym aplikacjom zmianę tabeli źródła danych, dla której zdefiniowano pseudonim. Ta instrukcja nie jest obsługiwana w źródłach danych ODBC, Microsoft SQL Server i nierelacyjnych.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie SELECT względem odpowiedniej tabeli • uprawnienie CONTROL względem odpowiedniej tabeli
REVOKE	Odbiera uprawnienia względem pseudonimów i widoków stowarzyszonych, takie jak ALTER, DELETE, INDEX, INSERT, SELECT, UPDATE. Uprawnienia względem obiektów źródła danych muszą być odbierane osobno.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie CONTROL względem obiektu
SELECT	Wybiera wiersze z obiektu źródła danych (takiego jak tabela lub widok), dla którego zdefiniowano pseudonim.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie SELECT względem pseudonimu i uprawnienie SELECT względem odpowiedniego obiektu źródła danych • uprawnienie CONTROL względem odpowiedniego obiektu źródła danych
UPDATE	Aktualizuje wartości określonych pól w wierszach obiektu źródła danych (takiego jak tabela lub widok), dla którego zdefiniowano pseudonim.	<ul style="list-style-type: none"> • SYSADM lub DBADM • uprawnienie UPDATE względem pseudonimu i uprawnienie UPDATE względem odpowiedniego obiektu źródła danych • uprawnienie CONTROL względem odpowiedniego obiektu źródła danych

Uprawnienia względem pseudonimów używanych w zapytaniu są sprawdzane podczas wprowadzania zapytania do stowarzyszonej bazy danych. Uprawnienia względem obiektów

źródła danych, do którego odwołuje się pseudonim, są sprawdzane tylko wtedy, gdy zapytanie jest rzeczywiście przetwarzane w źródle danych.

W celu wybrania, wstawienia, aktualizacji lub usunięcia danych przy użyciu pseudonimu, identyfikator autoryzowanego użytkownika wysyłającego daną instrukcję musi mieć następujące uprawnienia:

- odpowiednie uprawnienia względem pseudonimu (aby stowarzyszona baza danych zaakceptowała żądanie),
- odpowiednie uprawnienia względem obiektu tabeli (aby źródło danych zaakceptowało żądanie).

Na przykład do aktualizacji źródła danych przy użyciu pseudonimu wymagane jest uprawnienie UPDATE względem pseudonimu i uprawnienie UPDATE względem odpowiedniego obiektu źródła danych.

Zadania pokrewne:

- “Odwoływanie się do obiektów źródeł danych za pośrednictwem pseudonimów w instrukcjach SQL” na stronie 202

Informacje pokrewne:

- “ALTER TABLE statement” w podręczniku *SQL Reference, Volume 2*
- “COMMENT statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE ALIAS statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE INDEX statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE VIEW statement” w podręczniku *SQL Reference, Volume 2*
- “Instrukcja DELETE” w podręczniku *SQL Reference, Volume 2*
- “DROP statement” w podręczniku *SQL Reference, Volume 2*
- “GRANT (Database Authorities) statement” w podręczniku *SQL Reference, Volume 2*
- “Instrukcja INSERT” w podręczniku *SQL Reference, Volume 2*
- “LOCK TABLE statement” w podręczniku *SQL Reference, Volume 2*
- “REVOKE (Table, View, or Nickname Privileges) statement” w podręczniku *SQL Reference, Volume 2*
- “Instrukcja SELECT” w podręczniku *SQL Reference, Volume 2*
- “Instrukcja UPDATE” w podręczniku *SQL Reference, Volume 2*
- “ALTER NICKNAME statement” w podręczniku *SQL Reference, Volume 2*

Dostęp do nowych obiektów źródeł danych

Od czasu do czasu konieczne będzie uzyskanie dostępu do obiektów źródeł danych, dla których nie zdefiniowano pseudonimów. Mogą to być nowe obiekty dodane do źródła danych, na przykład nowo utworzony widok. Mogą to być także obiekty, które istniały podczas początkowej konfiguracji serwera stowarzyszonego, ale nie zostały na nim zarejestrowane. W każdym przypadku obiekty takie są dla serwera stowarzyszonego obiektami nowymi. W celu uzyskania dostępu do tych nowych obiektów należy utworzyć dla nich pseudonimy za pomocą instrukcji CREATE NICKNAME.

Wymagania wstępne:

W systemie stowarzyszonym musi być skonfigurowany dostęp do źródła danych.

W stowarzyszonej bazie danych musi istnieć definicja serwera źródła danych, na którym rezyduje dany obiekt. Do tworzenia definicji serwera służy instrukcja CREATE SERVER.

Wstawianie, aktualizowanie lub usuwanie danych za pośrednictwem pseudonimu wymaga spełnienia wszystkich wymienionych niżej warunków:

- identyfikator autoryzowanego użytkownika wysyłającego instrukcję musi mieć uprawnienie SELECT, INSERT, UPDATE lub DELETE względem pseudonimu (aby stowarzyszona baza danych zaakceptowała żądanie);
- identyfikator użytkownika w źródle danych musi mieć uprawnienie SELECT, INSERT, UPDATE lub DELETE względem odpowiedniego obiektu tabeli (aby źródło danych zaakceptowało żądanie);
- identyfikator użytkownika w źródle danych musi być odwzorowany na identyfikator autoryzowanego użytkownika na serwerze stowarzyszonym za pośrednictwem odwzorowania użytkownika.

Do korzystania z instrukcji CREATE NICKNAME wymagane jest jedno z poniższych uprawnień:

- SYSADM lub DBADM
- uprawnienie IMPLICIT_SCHEMA względem stowarzyszonej bazy danych, o ile nie istnieje jawna lub niejawna nazwa schematu pseudonimu
- uprawnienie CREATEIN względem schematu, o ile nazwa schematu pseudonimu istnieje

Procedura:

Instrukcja CREATE NICKNAME różni się nieznacznie dla relacyjnych i nierelacyjnych źródeł danych.

Składnia instrukcji CREATE NICKNAME odnoszącej się do relacyjnych źródeł danych jest następująca:

```
CREATE NICKNAME nazwa_pseudonimu FOR nazwa_serwera."zdalny_schemat"."nazwa_obiektu"
  OPTIONS (lista_opcji)
```

nazwa_pseudonimu

Unikalna nazwa pseudonimu dla obiektu źródła danych.

Pseudonim składa się z dwóch części: nazwy schematu i nazwy pseudonimu. Jeśli podczas tworzenia pseudonimu zostanie pominięta nazwa schematu, jako nazwa schematu dla pseudonimu zostanie użyty identyfikator autoryzowanego użytkownika tworzącego pseudonim. Maksymalna długość pseudonimu to 128 znaków.

FOR *nazwa_serwera*."*schemat_zdalny*".*"nazwa_obiektu"*

Trzyczęściowy identyfikator zdalnego obiektu źródła danych. Jeśli źródło danych nie obsługuje schematów, należy pominąć nazwę schematu w instrukcji CREATE NICKNAME.

- *nazwa_serwera* to nazwa przypisana serwerowi źródła danych w instrukcji CREATE SERVER.
- *schemat_zdalny* to nazwa schematu zdalnego, do którego należy obiekt.
- *nazwa_obiektu* to nazwa zdalnego obiektu, do którego ma być uzyskiwany dostęp.

OPTIONS (*lista_opcji*)

Informacje dotyczące pseudonimu, umożliwiające kompilatorowi SQL i opakowaniu efektywne przetwarzanie zapytań.

W przypadku niektórych nierelacyjnych źródeł danych składnia instrukcji CREATE NICKNAME jest następująca:

```
CREATE NICKNAME nazwa_pseudonimu lista_definicji_kolumn
  FOR SERVER nazwa_serwera
  OPTIONS (lista_opcji)
```

nazwa_pseudonimu

Unikalna nazwa pseudonimu dla obiektu źródła danych.

Pseudonim składa się z dwóch części: nazwy schematu i nazwy pseudonimu. Jeśli podczas tworzenia pseudonimu zostanie pominięta nazwa schematu, jako nazwa schematu dla pseudonimu zostanie użyty identyfikator autoryzowanego użytkownika tworzącego pseudonim. Maksymalna długość pseudonimu to 128 znaków.

lista_definicji_kolumn

Lista kolumn pseudonimu i ich typów danych.

FOR SERVER *nazwa_serwera*

Lokalna nazwa utworzona dla zdalnego serwera w instrukcji CREATE SERVER definiującej informacje o serwerze.

OPTIONS (*lista_opcji*)

Informacje dotyczące pseudonimu, umożliwiające kompilatorowi SQL i opakowaniu efektywne przetwarzanie zapytań.

Zadania pokrewne:

- “Praca z pseudonimami” na stronie 100

Informacje pokrewne:

- “CREATE NICKNAME statement” w podręczniku *SQL Reference, Volume 2*

Dostęp do źródeł danych przy użyciu sesji tranzytowych

Korzystając ze specjalnego trybu działania, zwanego *tranzytem*, można wprowadzać instrukcje SQL bezpośrednio do źródeł danych. Instrukcje SQL muszą spełniać wymogi dialektu SQL używanego w odpowiednim źródle danych. Sesje tranzytowe są przydatne, jeśli nie można wykonać danej operacji za pomocą interfejsu DB2 SQL/API. Na przykład w ramach sesji tranzytowej można utworzyć procedurę, indeks lub wykonywać zapytania w rodzimym dialekcie źródła danych.

Źródła danych obsługujące tranzyt obsługują obecnie wprowadzanie w sesjach tranzytowych jedynie instrukcji w języku SQL. Możliwe, że w przyszłości źródła danych będą obsługiwały w ramach sesji tranzytowej także języki inne niż SQL.

Sesje tranzytowe mogą także posłużyć do wykonywania działań, które nie są obsługiwane w języku SQL, na przykład niektórych zadań administracyjnych. Nie można jednak w ramach sesji tranzytowych wykonywać wszystkich zadań administracyjnych. To, jakie zadania administracyjne można wykonywać, zależy od źródła danych. Na przykład dla źródła DB2 UDB można uruchomić używany przez to źródło program narzędziowy do zbierania statystyk, ale nie można uruchomić ani zatrzymać zdalnej bazy danych.

W sesji tranzytowej niemożliwe jest wykonywanie zapytań odwołujących się jednocześnie do wielu źródeł danych. Sesję tranzytową otwiera się za pomocą komendy SET PASSTHRU. Komenda SET PASSTHRU RESET powoduje zamknięcie sesji tranzytowej. Użycie zamiast komendy SET PASSTHRU RESET komendy SET PASSTHRU spowoduje zamknięcie bieżącej sesji tranzytowej i otwarcie nowej sesji.

W odniesieniu do kursora zdefiniowanego w sesji tranzytowej można użyć składni WITH HOLD. Jeśli jednak źródło danych nie obsługuje tej składni, próba jej użycia (z instrukcją COMMIT) spowoduje wyświetlenie komunikatu o błędzie.

Sesje tranzytowe nie obsługują nierelacyjnych źródeł danych.

Pojęcia pokrewne:

- “Sesje tranzytowe” na stronie 10
- “Bezpośrednie kierowanie zapytań do źródeł danych w sesji tranzytowej” na stronie 213

Zadania pokrewne:

- “Praca z pseudonimami” na stronie 100

Informacje pokrewne:

- “SET PASSTHRU statement” w podręczniku *SQL Reference, Volume 2*

Uzyskiwanie dostępu do danych heterogenicznych za pośrednictwem widoków stowarzyszonych

Widok stowarzyszony jest to widok w stowarzyszonej bazie danych, którego tabele podstawowe znajdują się w zdalnych źródłach danych. Widok stowarzyszony odwołuje się do tabel podstawowych za pomocą pseudonimów, a nie nazw tabel w źródłach danych. Podczas wykonywania zapytania dotyczącego widoku stowarzyszonego dane są pobierane ze zdalnego źródła danych. Działanie tworzenia w stowarzyszonej bazie danych widoku danych ze źródła danych nazywa się niekiedy “tworzeniem widoku dla pseudonimu”. Jest to uzasadnione tym, że podczas tworzenia widoku instrukcje odwołują się do pseudonimów, a nie do źródeł danych.

Takie widoki zapewniają wysoki stopień niezależności danych w globalnie zintegrowanej bazie danych, podobnie jak widoki zdefiniowane dla wielu tabel lokalnych w scentralizowanych menedżerach relacyjnych baz danych.

Wymagania wstępne:

Do korzystania z instrukcji CREATE VIEW wymagane jest jedno z poniższych uprawnień:

- SYSADM lub DBADM
- dla każdego pseudonimu w dowolnej instrukcji pełnej selekcji:
 - CONTROL lub SELECT względem odpowiedniej tabeli lub widoku
 - jednego z poniższych uprawnień:
 - IMPLICIT_SCHEMA względem stowarzyszonej bazy danych, o ile nie istnieje jawna lub niejawna nazwa schematu widoku
 - CREATEIN względem schematu, o ile nazwa schematu widoku odwołuje się do istniejącego schematu

Uprawnienia do odpowiednich obiektów w źródłach danych nie są brane pod uwagę podczas definiowania widoku odwołującego się do pseudonimów w stowarzyszonej bazie danych.

Ograniczenia:

| Widoki stowarzyszone utworzone za pomocą instrukcji UNION ALL są przeznaczone tylko do odczytu.

| Widoki stowarzyszone obejmujące więcej niż jeden pseudonim w klauzuli FROM są widokami przeznaczonymi tylko do odczytu.

| Widoki stowarzyszone obejmujące więcej niż jeden pseudonim w klauzuli FROM mogą być widokami przeznaczonymi tylko do odczytu.

- Jeśli pseudonim w klauzuli FROM odnosi się do nierelacyjnego źródła danych, widok stowarzyszony jest przeznaczony tylko do odczytu.

- Jeśli podczas tworzenia widoku stowarzyszonego zostaną włączone inne pseudonimy w predykatkach lub w podzapytaniach, można będzie aktualizować ten widok.

Procedura:

Widok stowarzyszony tworzy się przy użyciu instrukcji CREATE VIEW.

Wymagane uprawnienia do tabeli lub widoku w źródle danych, do którego odwołuje się pseudonim, są sprawdzane podczas przetwarzania zapytania. Identyfikator autoryzowanego użytkownika wydającego instrukcję może być odwzorowany na inny zdalny identyfikator autoryzowanego użytkownika za pomocą odwzorowania użytkownika.

Tworzenie widoku stowarzyszonego łączącego podobne dane z kilku obiektów źródeł danych:

Załóżmy, że dane klientów pewnej firmy znajdują się na trzech odrębnych serwerach: w Europie, Azji i w Ameryce Południowej. Na serwerze europejskim dane znajdują się w tabeli Oracle. Pseudonim tej tabeli to ORA_EU_CUST. Dane klientów azjatyckich przechowywane są w tabeli Sybase. Pseudonim tej tabeli to SYB_AS_CUST. Dane klientów południowoamerykańskich są zapisane w tabeli Informix. Pseudonim tej tabeli to INFMX_SA_CUST. Każda z tabel zawiera kolumnę z numerem klienta (CUST_NO), nazwą klienta (CUST_NAME), numerem produktu (PROD_NO) i wielkością zamówienia (QUANTITY). Składnia instrukcji tworzącej na podstawie tych trzech pseudonimów widok łączący wszystkie dane klientów jest następująca:

```
CREATE VIEW FV1
AS SELECT * FROM ORA_EU_CUST
UNION ALL
SELECT * FROM SYB_AS_CUST
UNION ALL
SELECT * FROM INFMX_SA_CUST
```

Łączenie danych w celu utworzenia widoku stowarzyszonego:

Załóżmy, że dane klientów przechowywane są na jednym serwerze, a dane o sprzedaży na drugim. Dane klientów znajdują się w tabeli bazy danych Oracle. Pseudonim tej tabeli to ORA_EU_CUST. Dane dotyczące sprzedaży znajdują się w tabeli bazy danych Sybase. Pseudonim tej tabeli to SYB_SALES. Zależy nam na dopasowaniu informacji o klientach do informacji o zakupach dokonanych przez klientów. W każdej z tabel znajduje się kolumna z numerem klienta (CUST_NO). Składnia instrukcji tworzącej na podstawie dwóch pseudonimów widok stowarzyszony łączący te dane jest następująca:

```
CREATE VIEW FV4
AS SELECT A.CUST_NO, A.CUST_NAME, B.PROD_NO, B.QUANTITY
FROM ORA_EU_CUST A, SYB_SALES B
WHERE A.CUST_NO=B.CUST_NO
```

Zadania pokrewne:

- “Tworzenie i używanie widoków stowarzyszonych” na stronie 206
- “Praca z pseudonimami” na stronie 100

Informacje pokrewne:

- “CREATE VIEW statement” w podręczniku *SQL Reference, Volume 2*

Tworzenie pseudonimu dla pseudonimu

Sporadycznie konieczne może być utworzenie pseudonimu dla pseudonimu.

Procedura:

Załóżmy, że w sieci działa serwer stowarzyszony z systemem AIX[®] i serwer stowarzyszony z systemem Windows. Pożądane jest uzyskiwanie dostępu do arkuszy kalkulacyjnych programu Excel z obu serwerów stowarzyszonych. Jednak opakowanie dla programu Excel jest obsługiwane tylko na serwerach stowarzyszonych działających pod kontrolą systemu Windows. W celu uzyskania dostępu do arkusza programu Excel z serwera stowarzyszonego w systemie AIX należy wykonać następujące czynności:

1. Na serwerze stowarzyszonym w systemie Windows zainstaluj program DB2 Information Integrator.
2. Na serwerze stowarzyszonym w systemie Windows skonfiguruj dostęp do źródeł danych programu Excel.
3. Na serwerze stowarzyszonym w systemie Windows utwórz pseudonim dla arkusza programu Excel.
4. Na serwerze stowarzyszonym w systemie AIX zainstaluj program DB2 Information Integrator.
5. Na serwerze stowarzyszonym w systemie AIX skonfiguruj dostęp do źródeł danych z rodziny DB2.
6. Na serwerze stowarzyszonym w systemie AIX utwórz pseudonim dla pseudonimu arkusza programu Excel z serwera stowarzyszonego w systemie Windows.

Selekcja danych w systemie stowarzyszonym

Niektóre typy rozproszonych żądań używanych w systemie stowarzyszonym to:

- zapytania do jednego zdalnego źródła danych,
- zapytania do lokalnego źródła danych i do zdalnego źródła danych,
- zapytania do wielu zdalnych źródeł danych,
- zapytania do kombinacji zdalnych i lokalnych źródeł danych.

Aby wybrać dane ze źródeł danych (dokonać selekcji), w instrukcji SELECT należy użyć pseudonimów obiektów źródeł danych.

Wymagania wstępne:

Selekcja danych za pośrednictwem pseudonimu wymaga spełnienia wszystkich wymienionych niżej warunków:

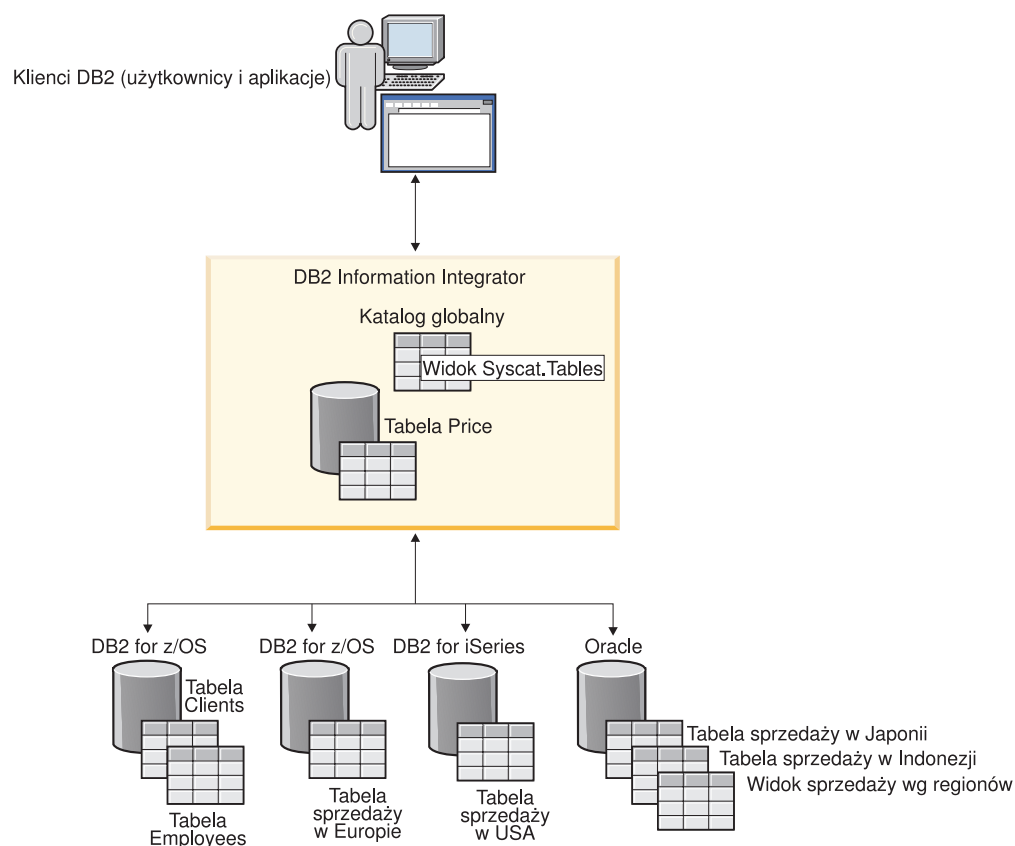
- identyfikator autoryzowanego użytkownika musi mieć uprawnienie SELECT względem pseudonimu (aby stowarzyszona baza danych zaakceptowała żądanie).
- identyfikator użytkownika w źródle danych musi mieć uprawnienie SELECT względem obiektu odpowiedniej tabeli (aby źródło danych zaakceptowało żądanie).
- identyfikator użytkownika w źródle danych musi być odwzorowany na identyfikator autoryzowanego użytkownika na serwerze stowarzyszonym za pośrednictwem odwzorowania użytkownika.

Procedura:

Stowarzyszona baza danych jest lokalnym źródłem danych. Obiektami lokalnymi są tabele i widoki w stowarzyszonej bazie danych. Dla tych obiektów nie tworzy się pseudonimów; w instrukcjach SELECT używa się rzeczywistych nazw obiektów.

Zdalne źródła danych to: inne instancje bazy danych DB2 dla systemów Linux, UNIX i Windows na serwerze stowarzyszonym, inne instancje bazy danych DB2 dla systemów Linux, UNIX i Windows na innym serwerze oraz źródła danych inne niż bazy danych DB2 dla systemów Linux, UNIX i Windows. Obiekty rezydujące w zdalnych źródłach danych to obiekty zdalne.

Załóżmy, że serwer stowarzyszony został skonfigurowany do korzystania ze źródeł danych DB2 for OS/390, DB2 for iSeries i Oracle. W każdym źródle danych przechowywana jest tabela z informacjami o sprzedaży. Taką konfigurację przedstawia poniższy rysunek.



Rysunek 3. Przykładowy system stowarzyszony ze źródłami danych DB2 i Oracle

Tabela z danymi o sprzedaży zawiera kolumny z numerem klienta (CUST_NO), zamówioną ilością (QUANTITY) i numerem zamówionego produktu (PROD_NO). Ponadto w stowarzyszonej bazie danych znajduje się tabela lokalna z informacjami o cenach produktów. Tabela z cenami zawiera kolumny z numerem produktu (PROD_NO) i bieżącą ceną (PRICE).

Pseudonimy dla obiektów zdalnego źródła danych są przechowywane w tabelach SYSCAT.TABLES, co przedstawiono na poniższym rysunku. Kolumna TYPE określa typ obiektu, na przykład pseudonim (N), tabela lokalna (T) lub widok (V).

Informacje o źródle danych

Nazwa obiektu źródła danych	Typ obiektu	Położenie
PRICES	Tabela lokalna	Stowarzyszona baza danych DB2
EUROPE_SALES	Tabela zdalna	Baza danych DB2 for z/OS and OS/390
US_SALES	Tabela zdalna	Baza danych DB2 for iSeries
JAPAN_SALES	Tabela zdalna	Baza danych Oracle
SALES_BY_REGION	Widok zdalny	Baza danych Oracle

Tabele SYSCAT.TABLES

NAZWA_TAB	TYP
PRICES	T
FED_PRICES	N
Z_EU_SALES	N
iS_US_SALES	N
ORA_JAPANSALES	N
ORA_REGIONSALES	N
.....	

Rysunek 4. Tabele i pseudonimy dla przykładowych zapytań

Poniżej przedstawiono przykłady instrukcji SELECT odwołujących się do opisanego powyżej przykładowego systemu stowarzyszonego.

Przykład: zapytanie do jednego źródła danych:

Tabela Z_EU_SALES zawiera zestawienie produktów zamówionych przez klientów z obszaru Europy. Zawiera ona także zamówioną ilość produktu w ramach każdej sprzedaży. Poniższe zapytanie wykorzystuje instrukcję SELECT z klauzulą ORDER BY w celu wyświetlenia listy transakcji sprzedaży w Europie i sortuje tę listę według numeru klienta:

```
SELECT CUST_NO, PROD_NO, QUANTITY
FROM Z_EU_SALES
ORDER BY CUST_NO
```

Przykład: łączenie lokalnego źródła danych i zdalnego źródła danych:

Tabela PRICES znajduje się w stowarzyszonej bazie danych. Zawiera ona listę cen wszystkich sprzedawanych produktów. Z tej tabeli lokalnej należy wybrać ceny odpowiadające produktom wymienionym w tabeli Z_EU_SALES. Ponadto należy posortować tabelę wynikową według numeru klienta.

```
SELECT sales.CUST_NO, sales.PROD_NO, sales.QUANTITY
FROM Z_EU_SALES sales, PRICES
WHERE sales.PROD_NO=PRICES.PROD_NO
ORDER BY sales.CUST_NO
```

Przykład: zapytanie do wielu zdalnych źródeł danych:

Załóżmy, że konieczne jest zebranie informacji o wszystkich transakcjach sprzedaży z każdego regionu i uporządkowanie tabeli wynikowej według numeru produktu.

```
WITH GLOBAL_SALES (Customer, Product, Quantity) AS
(SELECT CUST_NO, PROD_NO, QUANTITY FROM Z_EU_SALES
UNION ALL
SELECT CUST.NO, PROD.NO, QUANTITY FROM iS_US_SALES
UNION ALL
SELECT CUST.NO, PROD.NO, QUANTITY FROM ORA_JAPANSALES
SELECT Customer, Product, Quantity
FROM GLOBAL_SALES
ORDER BY Product
```

Przyjmijmy, że w źródle danych Oracle znajduje się widok, zawierający dane o sprzedaży w Japonii i Indonezji. Pseudonim dla tego widoku nazywa się ORA_SALESREGION. Informacje z tego widoku należy połączyć z danymi o sprzedaży w Stanach Zjednoczonych i wyświetlić ceny produktów obok każdej transakcji sprzedaży.

```
SELECT us_jpn_ind.CUST_NO, us_jpn_ind.PROD_NO,
us_jpn_ind.QUANTITY, us_jpn_ind.QUANTITY*PRICES.PRICE
AS SALEPRICE FROM
(SELECT CUST_NO, PROD_NO, QUANTITY
```

```

FROM ORA_SALESREGION
UNION ALL
SELECT CUST_NO, PROD_NO, QUANTITY
FROM iS_US_SALES us ) us_jpn_ind, PRICES
WHERE us_jpn_ind.PROD_NO = PRICES.PROD_NO
ORDER BY SALEPRICE DESC

```

Informacje pokrewne:

- “Instrukcja SELECT” w podręczniku *SQL Reference, Volume 2*

Modyfikowanie danych w systemie stowarzyszonym

W systemie stowarzyszonym można używać instrukcji INSERT, UPDATE i DELETE w odniesieniu do pseudonimów obiektów źródeł danych. Poniższe sekcje zawierają przykłady wykonywania takich operacji.

Wstawianie danych do obiektów źródeł danych

W celu wstawienia danych do źródeł danych w instrukcji INSERT należy użyć pseudonimów obiektów ze źródeł danych.

Wymagania wstępne:

Wstawianie danych za pośrednictwem pseudonimu wymaga spełnienia wszystkich wymienionych niżej warunków:

- identyfikator autoryzowanego użytkownika musi mieć uprawnienie INSERT względem pseudonimu (aby stowarzyszona baza danych zaakceptowała żądanie);
- identyfikator użytkownika w źródle danych musi mieć uprawnienie INSERT względem obiektu odpowiedniej tabeli (aby źródło danych zaakceptowało żądanie);
- identyfikator użytkownika w źródle danych musi być odwzorowany na identyfikator autoryzowanego użytkownika na serwerze stowarzyszonym za pośrednictwem odwzorowania użytkownika.

Ograniczenia:

W systemach stowarzyszonych nie są obsługiwane operacje INSERT w odniesieniu do niektórych źródeł danych, patrz sekcja Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym. Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym.

Procedura:

Niech tabela w bazie danych Informix składa się z dwóch kolumn. Pierwsza kolumna zawiera dane typu INTEGER, a druga dane typu VARCHAR (maksymalnie 20 znaków). Dla tej tabeli na serwerze stowarzyszonym zarejestrowano pseudonim *infx_tabela_nn*.

Korzystając z pseudonimu *infx_tabela_nn*, można wydawać instrukcje INSERT, UPDATE i DELETE dotyczące tabeli w bazie danych Informix. Na przykład instrukcja wstawiająca nowy wiersz danych w tej tabeli ma następującą postać:

```
INSERT INTO db2user1.infx_table_nn VALUES(1, 'Jacek')
```

Pojęcia pokrewne:

- “Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym” na stronie 96

Zadania pokrewne:

- “Selekcja danych w systemie stowarzyszonym” na stronie 110
- “Aktualizowanie danych w obiektach źródła danych” na stronie 114
- “Usuwanie danych z obiektów źródeł danych” na stronie 115

Informacje pokrewne:

- “Instrukcja INSERT” w podręczniku *SQL Reference, Volume 2*

Aktualizowanie danych w obiektach źródła danych

W celu aktualizacji danych w źródłach danych w instrukcji UPDATE należy użyć pseudonimów obiektów ze źródeł danych.

Wymagania wstępne:

Aktualizowanie danych za pośrednictwem pseudonimu wymaga spełnienia wszystkich wymienionych niżej warunków:

- identyfikator autoryzowanego użytkownika wysyłającego instrukcję musi mieć uprawnienie UPDATE względem pseudonimu (aby stowarzyszona baza danych zaakceptowała żądanie);
- identyfikator użytkownika w źródle danych musi mieć uprawnienie UPDATE względem odpowiedniego obiektu tabeli (aby źródło danych zaakceptowało żądanie);
- identyfikator użytkownika w źródle danych musi być odwzorowany na identyfikator autoryzowanego użytkownika na serwerze stowarzyszonym za pośrednictwem odwzorowania użytkownika.

Ograniczenia:

W systemach stowarzyszonych nie są obsługiwane operacje UPDATE w odniesieniu do niektórych źródeł danych, patrz sekcja Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym. Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym.

Procedura:

Niech tabela w bazie danych Informix składa się z dwóch kolumn. Pierwsza kolumna zawiera dane typu INTEGER, a druga dane typu VARCHAR (maksymalnie 20 znaków). Dla tej tabeli na serwerze stowarzyszonym zarejestrowano pseudonim *infx_tabela_nn*.

Korzystając z pseudonimu *infx_tabela_nn*, można wydawać instrukcje INSERT, UPDATE i DELETE dotyczące tabeli w bazie danych Informix. Na przykład instrukcja aktualizująca wiersz danych w tej tabeli ma następującą postać:

```
UPDATE db2user1.infx_table_nn SET c2='Maria' WHERE c1=2
```

Pojęcia pokrewne:

- “Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym” na stronie 96

Zadania pokrewne:

- “Selekcja danych w systemie stowarzyszonym” na stronie 110
- “Wstawianie danych do obiektów źródeł danych” na stronie 113
- “Usuwanie danych z obiektów źródeł danych” na stronie 115

Informacje pokrewne:

- “Instrukcja UPDATE” w podręczniku *SQL Reference, Volume 2*

Usuwanie danych z obiektów źródeł danych

W celu usunięcia danych ze źródeł danych w instrukcji DELETE należy użyć pseudonimów obiektów ze źródeł danych.

Wymagania wstępne:

Usuwanie danych za pośrednictwem pseudonimu wymaga spełnienia wszystkich wymienionych niżej warunków:

- identyfikator autoryzowanego użytkownika wysyłającego instrukcję musi mieć uprawnienie DELETE względem pseudonimu (aby stowarzyszona baza danych zaakceptowała żądanie);
- identyfikator użytkownika w źródle danych musi mieć uprawnienie DELETE względem odpowiedniego obiektu tabeli (aby źródło danych zaakceptowało żądanie);
- identyfikator użytkownika w źródle danych musi być odwzorowany na identyfikator autoryzowanego użytkownika na serwerze stowarzyszonym za pośrednictwem odwzorowania użytkownika.

Ograniczenia:

W systemach stowarzyszonych nie są obsługiwane operacje DELETE w odniesieniu do niektórych źródeł danych, patrz sekcja Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym.

Procedura:

Niech tabela w bazie danych Informix składa się z dwóch kolumn. Pierwsza kolumna zawiera dane typu INTEGER, a druga dane typu VARCHAR (maksymalnie 20 znaków). Dla tej tabeli na serwerze stowarzyszonym zarejestrowano pseudonim *infx_tabela_nn*.

Korzystając z pseudonimu *infx_tabela_nn*, można wydawać instrukcje INSERT, UPDATE i DELETE dotyczące tabeli w bazie danych Informix. Na przykład instrukcja usuwająca wiersz danych z tej tabeli ma następującą postać:

```
DELETE FROM infx_tabela_nn WHERE c1=3
```

Pojęcia pokrewne:

- “Ograniczenia dotyczące operacji INSERT, UPDATE i DELETE w systemie stowarzyszonym” na stronie 96

Zadania pokrewne:

- “Selekcja danych w systemie stowarzyszonym” na stronie 110
- “Wstawianie danych do obiektów źródeł danych” na stronie 113
- “Aktualizowanie danych w obiektach źródła danych” na stronie 114

Informacje pokrewne:

- “Instrukcja DELETE” w podręczniku *SQL Reference, Volume 2*

Semantyki przypisywania danych w systemie stowarzyszonym

Podczas przypisywania danych do kolumny pseudonimu typ danych może ulec zmianie zgodnie z regułami przypisywania używanymi przez program DB2 Information Integrator. Aby uzyskać wyniki zgodne z oczekiwaniami, należy poznać te reguły przypisywania.

Reguły określania docelowego typu danych dla przypisania do kolumny pseudonimu to:

- Określenie lokalnego typu źródłowego: lokalny typ źródłowy jest określony przez lokalny typ kolumny i lokalny typ wyniku wyrażenia. Jeśli wartość źródłowa jest stałą, lokalny typ źródłowy jest taki sam, jak typ stałej.
- Określenie typu docelowego:
 - Jeśli źródło przypisania nie ma typu, tak jak znaczniki parametrów i wartości puste, wówczas typ docelowy jest określony funkcją $\text{MIN}(\text{lokalny_typ_docelowy}, \text{zdalny_typ_docelowy})$, gdzie $\text{lokalny_typ_docelowy}$ to lokalny typ danych zaktualizowanej kolumny, a $\text{zdalny_typ_docelowy}$ to typ danych zaktualizowanej kolumny źródła danych. Typ $\text{zdalny_typ_docelowy}$ odwołuje się do domyślnego odwzorowania prostego typu danych zdalnej kolumny.
 - Jeśli źródło przypisania nie jest wartością pustą ani znacznikami parametrów, typ docelowy jest określony funkcją $\text{MIN}(\text{lokalny_typ_docelowy}, \text{zdalny_typ_docelowy}, \text{lokalny_typ_źródłowy})$.

Definicja funkcji $\text{MIN}(\text{typ1}, \text{typ2})$

- Typy typ1 i typ2 nie są identyczne.
- $\text{MIN}(\text{typ1}, \text{typ2}) = \text{MIN}(\text{typ2}, \text{typ1})$
- $\text{MIN}(\text{typ1}, \text{typ2}) = \text{zdalny_typ_docelowy}(\text{lokalny_typ_docelowy})$, gdy $\text{MIN}(\text{typ1}, \text{typ2}) = \text{DECIMAL}(0,0)$
- Typ BLOB jest zgodny tylko z typem BLOB, dlatego $\text{MIN}(\text{BLOB}(x), \text{BLOB}(y)) = \text{BLOB}(z)$, gdzie $z = \min(x, y)$
- Typy danych TIME i DATE nie są zgodne.
- Typy danych daty/godziny i łańcuchy danych są zgodne.
- W bazach danych obsługujących Unicode łańcuchy znaków i łańcuchy graficzne są zgodne.

Poniższa tabela przedstawia minima dla różnych kombinacji typów danych: liczbowego, łańcucha znaków, łańcucha graficznego oraz daty i godziny.

Tabela 10. Liczbowe typy danych

typ1	typ2	MIN(typ1, typ2)
SMALLINT	SMALLINT lub INTEGER lub BIGINT lub REAL lub DOUBLE	SMALLINT
INTEGER	BIGINT lub REAL lub DOUBLE	INTEGER
BIGINT	REAL lub DOUBLE	BIGINT
REAL	DOUBLE	REAL
DECIMAL(w,x)	SMALLINT	DECIMAL(p,0) gdzie $p = w - x$, jeśli $p < 5$; w przeciwnym razie SMALLINT
DECIMAL(w,x)	INTEGER	DECIMAL(p,0) gdzie $p = w - x$, jeśli $p < 11$; w przeciwnym razie INTEGER

Tabela 10. Liczbowe typy danych (kontynuacja)

typ1	typ2	MIN(typ1, typ2)
DECIMAL(w,x)	BIGINT	DECIMAL(p,0) gdzie $p=w-x$, jeśli $p < 19$; w przeciwnym razie BIGINT
DECIMAL(w,x)	DECIMAL(y,z)	DECIMAL(p,s) gdzie $p = \min(w,y) + \min(w-x,y-z)$, $s = \min(x,z)$
DECIMAL(w,x)	DOUBLE lub REAL	DECIMAL(w,x)

Poniższa tabela przedstawia minima dla różnych kombinacji typów danych łańcuchów znaków.

Tabela 11. Typy danych łańcuchów znaków

typ1	typ2	MIN(typ1, typ2)
CHAR(x)	CHAR(y) lub VARCHAR(y) lub LONG VARCHAR lub CLOB(y)	CHAR(z), gdzie $z = \min(x,y)$
VARCHAR(x)	VARCHAR(y) lub LONG VARCHAR lub CLOB(y)	VARCHAR(z), gdzie $z = \min(x,y)$
LONG VARCHAR	CLOB(y)	LONG VARCHAR gdy $x > 32700$, CLOB(x) gdy $x \leq 32700$
CLOB(x)	CLOB(y)	CLOB(z), gdzie $z = \min(x,y)$

Poniższa tabela przedstawia minima dla różnych kombinacji typów danych łańcuchów graficznych.

Tabela 12. Typy danych łańcuchów graficznych

typ1	typ2	MIN(typ1, typ2)
GRAPHIC(x)	GRAPHIC(y) lub VARGRAPHIC(y) lub LONG VARGRAPHIC lub DBCLOB(y)	GRAPHIC(z), gdzie $z = \min(x,y)$
VARGRAPHIC(x)	VARGRAPHIC(y) lub LONG VARGRAPHIC lub DBCLOB(y)	VARGRAPHIC(z), gdzie $z = \min(x,y)$
LONG VARGRAPHIC	DBCLOB(y)	LONG VARGRAPHIC, jeśli $x > 32700$; DBCLOB(x), jeśli $x \leq 32700$
DBCLOB(x)	DBCLOB(y)	DBCLOB(z), gdzie $z = \min(x,y)$

Poniższa tabela przedstawia minima dla różnych kombinacji typów danych daty i godziny.

Tabela 13. Typy danych daty i godziny

typ1	typ2	MIN(typ1, typ2)
DATE	TIMESTAMP	DATE
TIME	TIMESTAMP	TIME

Jeśli wstawiane dane typu CHAR są krótsze niż długość docelowa, w źródle danych są one dopełniane do szerokości kolumny.

Jeśli dane typu DATE lub TIME są wstawiane do zdalnej kolumny typu TIMESTAMP, w źródle danych są one dopełniane do szerokości kolumny.

Semantyki przypisań danych w systemie stowarzyszonym - przykłady

Tabela 14 przedstawia kilka przykładów zastosowań stowarzyszonych semantyk przypisań w zapytaniach o danym typie lokalnym i zdalnym.

Tabela 14. Przykłady semantyk przypisań

Typ lokalny	Typ zdalny	Zapytanie	Wygenerowane zapytanie zdalne
FLOAT	INTEGER	set c1=123,23	set c1=INTEGER(123,23)
INTEGER	FLOAT	set c1=123,23	set c1=INTEGER(123,23)
FLOAT	INTEGER	set c1=123	set c1=123
CHAR(10)	CHAR(20)	set c1='123'	set c1='123' - '123' jest typu VARCHAR(3) i jest najkrótszym ze wszystkich typów
CHAR(10)	CHAR(20)	set c1=char23col	set c1=CHAR(char23col, 10)
CHAR(10)	CHAR(20)	set c1=wyr1	<ul style="list-style-type: none"> • set c1=wyr1 -- jeśli wyrażenie wyr1 zwraca wynik typu char(n), gdzie n<=10 • set c1=CHAR(wyr1, 10) -- jeśli wyr1 zwraca wartość typu char(n), gdzie n>10
TIMESTAMP	DATE	set c1= current timestamp	set c1=DATE(current timestamp)

Rozdział 9. Monitorowanie systemu stowarzyszonego

W tym rozdziale opisano, w jaki sposób monitorować poprawność pseudonimów i serwerów oraz obrazy fragmentów zapytań.

Ten rozdział zawiera:

- “Indykatory poprawności dla stowarzyszonych pseudonimów i serwerów”
- “Aktywowanie stowarzyszonych indykatorów poprawności” na stronie 120
- “Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów” na stronie 121
- “Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów - przykład” na stronie 122
- “Monitorowanie obrazów stanu systemów stowarzyszonych - przegląd” na stronie 123
- “Monitorowanie fragmentów stowarzyszonych zapytań” na stronie 123
- “Monitorowanie obrazów stanu fragmentów zapytań stowarzyszonych - przykład” na stronie 124

Indykatory poprawności dla stowarzyszonych pseudonimów i serwerów

Indykatory poprawności w Centrum kontroli poprawności DB2[®] służą do monitorowania statusu stowarzyszonych pseudonimów i serwerów. Indykator poprawności dla pseudonimów to `db.fed_nicknames_op_status`. Indykator poprawności dla definicji serwerów to `db.fed_servers_op_status`. Stowarzyszone indykatory poprawności są instalowane podczas instalacji monitora poprawności.

Domyślnie Centrum kontroli poprawności nie aktywuje stowarzyszonych indykatorów poprawności. Indykatory te należy aktywować samodzielnie.

Gdy stan pseudonimu lub serwera jest inny niż normalny, indykatory poprawności generują alert. Wyniki monitorowania można przeglądać w Centrum kontroli poprawności lub przy użyciu wiersza komend.

Indykatory poprawności są obsługiwane przez serwery stowarzyszone działające w systemach operacyjnych AIX[®], HP-UX, Linux, Microsoft[®] Windows[®] i Solaris.

Tabela 15 przedstawia opisy indykatorów poprawności dla stowarzyszonych pseudonimów i serwerów.

Tabela 15. Indykatory poprawności pseudonimów i serwerów

Indykator poprawności	Opis
<code>db.fed_nicknames_op_status</code>	<p>Wskazuje zagregowany status poprawności wszystkich relacyjnych pseudonimów zdefiniowanych w bazie danych na serwerze stowarzyszonym DB2 UDB.</p> <p>Wysyła alerty, gdy pseudonim jest niepoprawny. Udostępnia szczegółowe informacje o niepoprawnych pseudonimach i zaleca działania do wykonania w celu ich naprawy.</p>

Tabela 15. Indykatory poprawności pseudonimów i serwerów (kontynuacja)

Indykator poprawności	Opis
db.fed_servers_op_status	Wskazuje zagregowany status poprawności wszystkich serwerów stowarzyszonych zdefiniowanych w bazie danych na serwerze stowarzyszonym DB2 UDB. Wysyła alerty, gdy serwer jest niedostępny. Dostarcza szczegółowe informacje o niedostępnych serwerach i zaleca działania do wykonania w celu ich naprawy.

Indykatory poprawności mogą analizować następujące źródła danych:

- produkty z rodziny DB2 (DRDA)
- Excel
- Informix®
- Microsoft SQL Server
- ODBC
- Oracle (NET8)
- Sybase (CTLIB)
- Pliki o strukturze tabeli
- Teradata
- XML (tylko pseudonimy elementu głównego)

Pojęcia pokrewne:

- “Introduction to the health monitor” w podręczniku *System Monitor Guide and Reference*

Zadania pokrewne:

- “Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów” na stronie 121
- “Aktywowanie stowarzyszonych indykatorów poprawności” na stronie 120

Informacje pokrewne:

- “db2hc - Start Health Center Command” w podręczniku *Command Reference*
- “db.fed_nicknames_op_status - Nickname Status health indicator” w podręczniku *System Monitor Guide and Reference*
- “db.fed_servers_op_status - Data Source Server Status health indicator” w podręczniku *System Monitor Guide and Reference*

Aktywowanie stowarzyszonych indykatorów poprawności

Aby możliwe było monitorowanie poprawności pseudonimów i serwerów, konieczne jest aktywowanie stowarzyszonych indykatorów poprawności.

Indykator poprawności dla pseudonimów to db.fed_nicknames_op_status. Indykator poprawności dla definicji serwerów to db.fed_servers_op_status.

Procedura:

Aby aktywować stowarzyszone indykatory poprawności, należy otworzyć Centrum kontroli poprawności DB2 i skonfigurowanie indykatorów poprawności.

Zadania pokrewne:

- “Configuring health indicators using Health Center” w podręczniku *System Monitor Guide and Reference*
- “Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów” na stronie 121

Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów

Monitorowanie statusu pseudonimu i serwera może pomóc w określeniu i rozwiązaniu problemów występujących w systemie stowarzyszonym. Status stowarzyszonych pseudonimów i serwerów można monitorować przy użyciu indykatorów poprawności w Centrum kontroli poprawności.

Wyniki monitorowania można przeglądać w Centrum kontroli poprawności lub przy użyciu wiersza komend. W celu rozwiązania problemów zidentyfikowanych przez indykatory poprawności należy użyć Centrum sterowania DB2 lub procesora wiersza komend DB2.

Wymagania wstępne:

- Należy sprawdzić, czy na serwerze stowarzyszonym DB2 zdefiniowano uprawnienia SELECT w odniesieniu do pseudonimów.
- Parametrowi konfiguracyjnemu menedżera bazy danych FEDERATED należy nadać wartość YES.
- Jeśli źródło danych wymaga uwierzytelniania, konieczne jest zdefiniowanie odwzorowania identyfikatora użytkownika Monitora poprawności na użytkownika tego źródła. Monitor poprawności użyje tego odwzorowania do połączenia ze źródłem danych.

Ograniczenia:

Indykatory poprawności nie mogą analizować stanu następujących źródeł danych:

- BioRS
- BLAST
- Documentum
- Entrez
- Extended Search
- HMMER
- Usługi WWW
- WebSphere Business Integration
- XML (pseudonimy elementów innych niż główny)

Procedura:

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Otwórz Centrum kontroli poprawności.
2. Otwórz Doradcę utrzymania poprawności, aby wyświetlić zalecenia dotyczące rozwiązania problemów z niepoprawnymi pseudonimami lub niedostępnymi serwerami.

Aby wykonać to zadanie z wiersza komend, należy użyć komendy GET HEALTH SNAPSHOT.

Pojęcia pokrewne:

- “Introduction to the health monitor” w podręczniku *System Monitor Guide and Reference*

- “Indykatory poprawności dla stowarzyszonych pseudonimów i serwerów” na stronie 119

Zadania pokrewne:

- “Resolving alerts using the Health Center” w podręczniku *System Monitor Guide and Reference*

Informacje pokrewne:

- “Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów - przykład” na stronie 122

Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów - przykład

Poniższy przykład przedstawia obraz stanu poprawności bazy danych o nazwie fedhi. Nazwy stowarzyszonych indykatorów poprawności to db.fed_nicknames_op_status i db.fed_servers_op_status. W tym przykładzie obydwa indykatory poprawności znajdują się w stanie normalnym. Stan "normalny" oznacza, że pseudonimy i serwery są poprawne.

GET HEALTH SNAPSHOT FOR DATABASE ON fedhi:

Obraz stanu poprawności bazy danych

Datownik obrazu stanu	= 02/10/2004 12:10:55.063004
Nazwa bazy danych	= FEDHI
Ścieżka do bazy danych	= C:\DB2\NODE0000\SQL00006\
Alias wejściowej bazy danych	= FEDHI
System operacyjny uruchomiony na serwerze bazy danych	= NT
Położenie bazy danych	= Lokalny
Najpoważniejszy alert dla bazy danych	= Uwaga

Indykatory poprawności:

Nazwa indykatora	= db.fed_servers_op_status
Wartość	= 0
Datownik oceny	= 02/10/2004 12:09:10.961000
Stan alertu	= Normalny
Nazwa indykatora	= db.fed_nicknames_op_status
Wartość	= 0
Datownik oceny	= 02/10/2004 12:09:10.961000
Stan alertu	= Normalny
Nazwa indykatora	= db.db_op_status
Wartość	= 0
Datownik oceny	= 02/10/2004 12:08:10.774000
Stan alertu	= Normalny
Nazwa indykatora	= db.sort_shrmem_util
Wartość	= 0
Jednostka	= %
Datownik oceny	= 02/10/2004 12:08:10.774000
Stan alertu	= Normalny
Nazwa indykatora	= db.spilled_sorts
Wartość	= 0
Jednostka	= %
Datownik oceny	= 02/10/2004 12:09:10.961000
Stan alertu	= Normalny

Pojęcia pokrewne:

- “Indykatory poprawności dla stowarzyszonych pseudonimów i serwerów” na stronie 119

Zadania pokrewne:

- “Monitorowanie poprawności stowarzyszonych pseudonimów i serwerów” na stronie 121

Monitorowanie obrazów stanu systemów stowarzyszonych - przegląd

Przy użyciu monitora obrazów stanu można w określonym czasie przechwytywać informacje o stowarzyszonych źródłach danych i dowolnych połączonych aplikacjach. Obrazy stanu są przydatne do określania statusu systemu stowarzyszonego. Przechwytywane w regularnych odstępach czasu obrazy te są także przydatne do obserwowania trendów i przewidywania potencjalnych problemów.

Pojęcia pokrewne:

- “Snapshot monitor” w podręczniku *System Monitor Guide and Reference*

Zadania pokrewne:

- “Monitorowanie fragmentów stowarzyszonych zapytań” na stronie 123

Informacje pokrewne:

- Rozdział 31, “Elementy monitora systemu stowarzyszonej bazy danych”, na stronie 307
- “Monitorowanie obrazów stanu fragmentów zapytań stowarzyszonych - przykład” na stronie 124

Monitorowanie fragmentów stowarzyszonych zapytań

Monitorowanie fragmentów zapytań pozwala ocenić działanie systemu stowarzyszonego. Aby poznać sposób przetwarzania zapytania przez system stowarzyszony, można sporządzić obraz stanu segmentów zdalnego zapytania.

Procedura:

Aby rozpocząć monitorowanie fragmentów zapytania, należy uruchomić komendę GET SNAPSHOT FOR DYNAMIC SQL ON <nazwa_bazy_danych>, gdzie *nazwa_bazy_danych* to nazwa lokalnej bazy danych na serwerze stowarzyszonym.

Pojęcia pokrewne:

- “Monitorowanie obrazów stanu systemów stowarzyszonych - przegląd” na stronie 123

Informacje pokrewne:

- “GET SNAPSHOT Command” w podręczniku *Command Reference*
- “Monitorowanie obrazów stanu fragmentów zapytań stowarzyszonych - przykład” na stronie 124

Monitorowanie obrazów stanu fragmentów zapytań stowarzyszonych - przykład

Poniższy przykład przedstawia dane wyjściowe obrazu stanu dynamicznego SQL dla fragmentu zapytania wysłanego do zdalnego źródła danych Oracle o nazwie ORACLE817. Zostały one wygenerowane w wyniku działania instrukcji GET SNAPSHOT FOR DYNAMIC SQL ON LOCAL_FEDERATED_DATABASE. Wyniki zawierają informacje dotyczące każdego zdalnego zapytania i wszystkich zapytań w pamięci podręcznej instrukcji. Ponieważ informacje o puli buforów nie dotyczą zapytań zdalnych, obraz stanu nie zbiera tych informacji.

Wynik w postaci obrazu dynamicznego SQL

```
Liczba wykonań                = 1
Liczba kompilacji              = 1
Najgorszy czas przygotowania (ms) = 215
Najlepszy czas przygotowania (ms) = 215
Wewnętrznie usunięte wiersze    = 0
Wewnętrznie wstawione wiersze  = 0
Przeczytane wiersze            = 25412
Wewnętrznie zaktualizowane wiersze = 0
Zapisane wiersze                = 25410
Sortowania instrukcji           = 0
Logiczne odczyty danych w buforach      = Nie zebrane
Fizyczne odczyty danych w buforach      = Nie zebrane
  Logiczne odczyty danych tymczasowych z puli buforów = Nie zebrane
  Fizyczne odczyty danych tymczasowych z puli buforów = Nie zebrane
  Logiczne odczyty indeksów z puli buforów = Nie zebrane
  Fizyczne odczyty indeksów z puli buforów = Nie zebrane
  Logiczne odczyty indeksów tymczasowych z puli buforów = Nie zebrane
  Fizyczne odczyty indeksów tymczasowych z puli buforów = Nie zebrane
Łączny czas wykonania (s.ms)           = 20.229786
Łączny czas pracy cpu użytkownika (s.ms) = 10.080000
Łączny czas pracy cpu systemu (s.ms)    = 0.520000
Tekst instrukcji = [ORACLE817]SELECT A0.C1,A0.C2 FROM ORA_T A0
                                WHERE A0.C3 = :H0
```

Pojęcia pokrewne:

- “Monitorowanie obrazów stanu systemów stowarzyszonych - przegląd” na stronie 123

Zadania pokrewne:

- “Monitorowanie fragmentów stowarzyszonych zapytań” na stronie 123

Informacje pokrewne:

- “GET SNAPSHOT Command” w podręczniku *Command Reference*

Rozdział 10. Obsługa Unicode dla stowarzyszonych źródeł danych

W tym rozdziale opisano obsługę kodowania Unicode w systemie stowarzyszonym.

Ten rozdział zawiera:

- “Obsługa Unicode w systemach stowarzyszonych”
- “Określanie strony kodowej klienta w celu obsługi kodowania Unicode w źródłach danych Microsoft SQL Server i ODBC” na stronie 127
- “Strony kodowe obsługiwane przez bazę danych Unicode dla opcji CODEPAGE opakowań MSSQL i ODBC” na stronie 128
- “Określanie strony kodowej pliku na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli” na stronie 128
- “Określanie strony kodowej na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli - przykład” na stronie 129
- “Błędy spowodowane różnicą wielkości zdalnego i stowarzyszonego punktu kodowego” na stronie 129

Obsługa Unicode w systemach stowarzyszonych

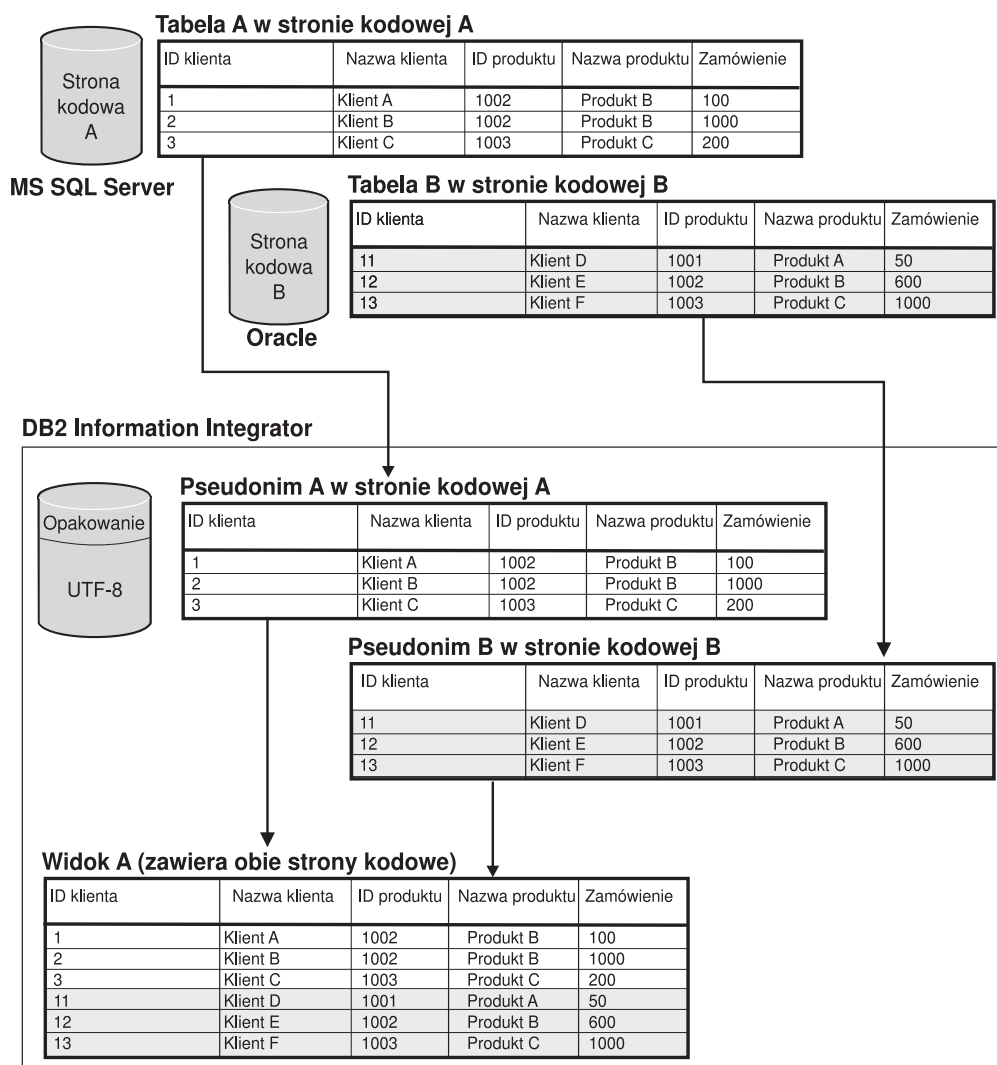
Opakowania relacyjne i nierelacyjne oraz funkcje zdefiniowane przez użytkownika mogą działać w bazach danych obsługujących kodowanie Unicode (bazy danych UTF-8). Baza danych Unicode udostępnia środowisko serwera stowarzyszonego, które jest niezależne od platformy. Baza danych Unicode może obsługiwać dane w różnych stronach kodowych przechowywane w różnych źródłach danych.

Opakowania i funkcje zdefiniowane przez użytkownika, które obsługują Unicode, to:

- Opakowania relacyjne
 - DRDA[®]
 - Informix[®]
 - MS SQL Server
 - ODBC
 - OLE DB
 - Oracle
 - Sybase
 - Teradata
- Opakowania nierelacyjne i funkcje zdefiniowane przez użytkownika
 - opakowanie BioRS
 - opakowanie BLAST
 - opakowanie Documentum
 - opakowanie Entrez
 - opakowanie Excel
 - opakowanie HMMER
 - opakowanie IBM[®] Lotus[®] Extended Search
 - funkcje zdefiniowane przez użytkownika KEGG
 - funkcje zdefiniowane przez użytkownika MQ
 - opakowanie pliku o strukturze tabeli
 - zdefiniowane przez użytkownika funkcje usług WWW
 - opakowanie usług WWW
 - opakowanie WebSphere[®] Business Integration

– opakowanie XML

Rys. 5 przedstawia firmę z oddziałami w różnych krajach. Każdy oddział przechowuje dane o klientach we własnych bazach danych i we własnej stronie kodowej. W bazie danych Microsoft® SQL Server dane są przechowywane w stronie kodowej A. W bazie danych Oracle dane są przechowywane w stronie kodowej B. Strony kodowe A i B obowiązują w różnych regionach geograficznych. Aby zintegrować dane z różnych regionów, dla stowarzyszonej bazy danych wybrano stronę kodową Unicode. Dzięki temu można połączyć tabele z różnych regionów, aby zobaczyć łączną liczbę zamówień.



Rysunek 5. Przykład zastosowania kodu Unicode

Zadania pokrewne:

- “Określanie strony kodowej klienta w celu obsługi kodowania Unicode w źródłach danych Microsoft SQL Server i ODBC” na stronie 127
- “Określanie strony kodowej pliku na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli” na stronie 128

Informacje pokrewne:

- “Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie NET8” na stronie 297

- “Strony kodowe obsługiwane przez bazę danych Unicode dla opcji CODEPAGE opakowań MSSQL i ODBC” na stronie 128
- “Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie NET8” na stronie 297
- “Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie Sybase” na stronie 298
- “Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie Sybase” na stronie 298
- “Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie ODBC” na stronie 299
- “Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie ODBC” na stronie 299
- “Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie Microsoft SQL Server” na stronie 300
- “Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie Microsoft SQL Server” na stronie 300
- “Określanie strony kodowej na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli - przykład” na stronie 129

Określanie strony kodowej klienta w celu obsługi kodowania Unicode w źródłach danych Microsoft SQL Server i ODBC

W celu zapewnienia poprawnej konwersji stron kodowych dla źródeł danych Microsoft SQL Server i ODBC konieczne jest określenie strony kodowej programu klienta, jeśli jest ona inna niż strona kodowa stowarzyszonej bazy danych.

Procedura:

Należy użyć instrukcji CREATE SERVER z opcją CODEPAGE o wartości równej stronie kodowej programu klienta.

Jeśli na przykład źródłem danych jest program Microsoft SQL Server, serwer stowarzyszony działa w systemie Windows i domyślne ustawienia narodowe systemu operacyjnego odpowiadają językowi japońskiemu (Shift-JIS), opcji serwera CODEPAGE należy nadać wartość 943 (Shift-JIS) lub 1202 (UTF-16LE).

Aby określić stronę kodową 1202 dla źródła danych Microsoft SQL Server o nazwie FEDSERVERW, należy użyć następującej instrukcji:

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

Jeśli źródłem danych jest program Microsoft SQL Server, serwer stowarzyszony działa w systemie UNIX a ustawienie AppCodePage lub IANAAppCodePage programu klienta DataDirect Connect to 6 (Shift-JIS), opcja serwera CODEPAGE musi mieć wartość 943 (Shift-JIS) lub 1208 (UTF-8).

Aby określić stronę kodową 1208 dla źródła danych Microsoft SQL Server o nazwie FEDSERVERU, należy użyć następującej instrukcji:

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

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Informacje pokrewne:

- “CREATE SERVER statement” w podręczniku *SQL Reference, Volume 2*
- Rozdział 21, “Opcje serwera w systemie stowarzyszonym”, na stronie 225
- “Strony kodowe obsługiwane przez bazę danych Unicode dla opcji CODEPAGE opakowań MSSQL i ODBC” na stronie 128

Strony kodowe obsługiwane przez bazę danych Unicode dla opcji CODEPAGE opakowań MSSQL i ODBC

Poprawne strony kodowe to takie, które obsługuje program DB2 Universal Database oraz te, które przedstawia Tabela 16.

Tabela 16. Strony kodowe obsługiwane przez bazę danych Unicode dla opcji CODEPAGE opakowań MSSQL i ODBC

Wartość opcji CODEPAGE	Opis
1200	Codepage1200 - UCS-2 (big-endian)
1202	Codepage1202 - UCS-2 (little-endian)
1208	Codepage1208 - UTF-8
1232	Codepage1232 - UTF-32 (big-endian)
1234	Codepage1234 - UTF-32 (little-endian)

Zadania pokrewne:

- “Określanie strony kodowej klienta w celu obsługi kodowania Unicode w źródłach danych Microsoft SQL Server i ODBC” na stronie 127

Informacje pokrewne:

- “Supported territory codes and code pages” w podręczniku *Administration Guide: Planning*

Określanie strony kodowej pliku na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli

W celu zapewnienia poprawnej konwersji stron kodowych dla źródeł danych w postaci pliku o strukturze tabeli konieczne jest określenie strony kodowej pliku, o ile jest ona inna niż strona kodowa stowarzyszonej bazy danych.

Ograniczenia:

Opcji CODEPAGE można używać tylko w stowarzyszonej bazie danych z obsługą Unicode.

Procedura:

Należy użyć instrukcji CREATE NICKNAME z opcją CODEPAGE odpowiadającą stronie kodowej danych w pliku o strukturze tabeli. Poprawne wartości to takie, które obsługuje program DB2 Universal Database. Wartością domyślną jest strona kodowa stowarzyszonej bazy danych programu DB2 Universal Database.

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Informacje pokrewne:

- “CREATE NICKNAME statement” w podręczniku *SQL Reference, Volume 2*
- “Supported territory codes and code pages” w podręczniku *Administration Guide: Planning*
- “Określanie strony kodowej na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli - przykład” na stronie 129
- “CREATE NICKNAME statement syntax - Table-structured file wrapper” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Określanie strony kodowej na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli - przykład

W pliku o nazwie DRUGDATA1.TXT znajdują się dane w stronie kodowej 943. Aby dla pliku o strukturze tabeli określić stronę kodową 943, należy użyć następującej instrukcji CREATE NICKNAME:

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

Zadania pokrewne:

- “Określanie strony kodowej pliku na potrzeby obsługi Unicode w źródłach danych w postaci plików o strukturze tabeli” na stronie 128

Błędy spowodowane różnicą wielkości zdalnego i stowarzyszonego punktu kodowego

Gdy między stowarzyszoną bazą danych i zdalnym źródłem danych występuje różnica wielkości punktów kodowych, zwrócone wyniki mogą być niepoprawne albo operacje wstawiania mogą się nie powieść.

Gdy punkt kodowy w stowarzyszonej bazie danych jest większy niż w zdalnym źródle danych, serwer stowarzyszony może obciąć dane wybierane ze zdalnego źródła danych. Dane są obcinane, gdy wynik konwersji łańcucha znaków ma więcej bajtów niż wielkość kolumny pseudonimu. Pozostałe bajty są puste. Można także wstawiać dane większe niż wielkość kolumny pseudonimu. Ten typ konwersji powiedzie się, gdy wynik konwersji będzie miał nie więcej bajtów niż wynosi wielkość zdalnej kolumny.

Gdy punkt kodowy w stowarzyszonej bazie danych jest mniejszy niż w zdalnym źródle danych, operacja wstawienia danych może się nie powieść. Operacja wstawienia nie powiedzie się, gdy wynik konwersji łańcucha znaków będzie miał więcej bajtów niż wielkość kolumny zdalnego źródła danych.

Aby opisane powyżej błędy nie występowały, wielkości punktów kodowych stowarzyszonej bazy danych i zdalnego źródła danych nie mogą zbyt różnić się.

Część 3. Wydajność

Rozdział 11. Strojenie wydajności w systemie stowarzyszonym

Problemy z wydajnością mogą występować w stowarzyszonej bazie danych, w źródłach danych, albo w obu tych miejscach jednocześnie. Przyczyną obniżenia wydajności może być wąskie gardło w stowarzyszonej bazie danych lub w źródle danych. W celu wyizolowania tych problemów konieczne jest strojenie stowarzyszonej bazy danych i źródeł danych prowadzące do uzyskania maksymalnej wydajności. W celu rozwiązania tych problemów konieczne może być dostrojenie zapytań, aplikacji, parametrów konfiguracyjnych i poziomu użycia sieci.

Niniejszy rozdział zawiera następujące sekcje:

- “Publikacje na temat wydajności systemów stowarzyszonych”
- “Strojenie przetwarzania zapytań”
- “Analiza przekazywania do źródła” na stronie 135
- “Globalna optymalizacja” na stronie 146
- “Elementy monitora systemu mające wpływ na wydajność” na stronie 154

Publikacje na temat wydajności systemów stowarzyszonych

Poniższe dokumenty opracowane przez firmę IBM zawierają szczegółowe informacje na temat strojenia wydajności:

- *Data Federation with IBM DB2 Information Integrator V8.1*, pod adresem <http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg247052.html?Open>
- “Using the federated database technology of IBM DB2 Information Integrator”, pod adresem <ftp://ftp.software.ibm.com/software/data/pubs/papers/iifed.pdf>
- “DB2 Information Integrator XML Wrapper Performance”, pod adresem <ftp://ftp.software.ibm.com/software/data/pubs/papers/db2iixmlwrapper.pdf>

Strojenie przetwarzania zapytań

Aby uzyskać dane ze źródeł danych, klienci (użytkownicy i aplikacje) wprowadzają do stowarzyszonej bazy danych zapytania w języku SQL programu DB2®. Następnie kompilator języka SQL w programie DB2 odczytuje z katalogu globalnego i opakowania źródła danych informacje, które pomogą przetworzyć zapytanie. Informacje te dotyczą połączenia ze źródłem danych, atrybutów serwera, odwzorowań, indeksu i statystyk pseudonimu.

W procesie kompilacji SQL zapytanie jest analizowane przez *optymalizator zapytań*. Kompilator opracowuje alternatywne strategie przetwarzania zapytania zwane *planami dostępu*. Plany dostępu mogą zakładać:

- przetworzenie zapytania przez źródła danych,
- przetworzenie zapytania przez serwer stowarzyszony,
- częściowe przetworzenie zapytania przez źródła danych i częściowe przetworzenie przez serwer stowarzyszony.

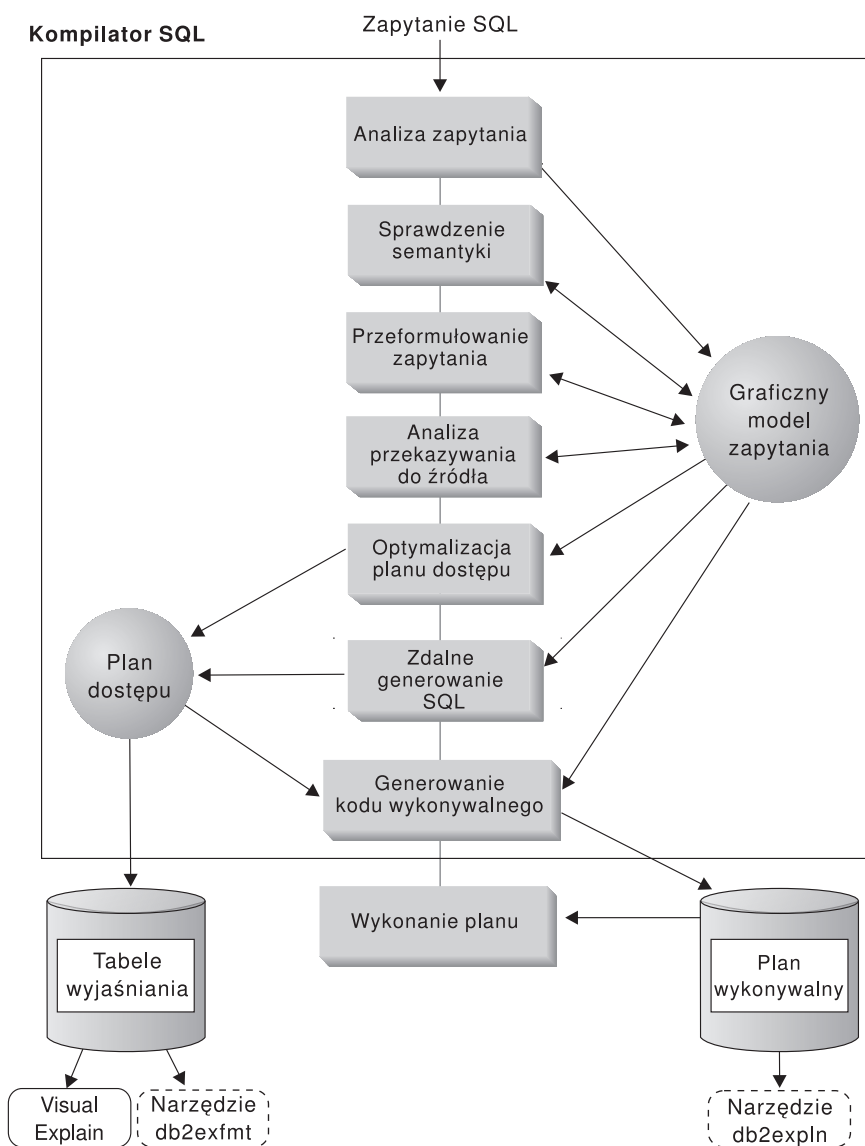
Program DB2 UDB ocenia plan dostępu przede wszystkim na podstawie informacji o możliwościach źródła danych i o samych danych. Odpowiednie informacje zawarte są w opakowaniu oraz w katalogu globalnym. Program DB2 UDB rozkłada zapytanie na segmenty nazywane *fragmentami zapytania*. Zazwyczaj efektywniejsze jest *przekazanie* fragmentu

zapytania do źródła danych, o ile źródło danych może go przetworzyć. Jednak optymalizator zapytań uwzględnia jeszcze inne czynniki, takie jak:

- objętość danych, które mają być przetworzone;
- szybkość przetwarzania danych przez źródło danych;
- objętość danych, które fragment zapytania ma zwrócić;
- przepustowość łącza komunikacyjnego.

Analiza przekazywania do źródła jest wykonywana tylko dla relacyjnych źródeł danych. Nierelacyjne źródła danych używają protokołu żądanie-odpowieź-kompensacja.

Poniższy rysunek ilustruje czynności wykonywane przez kompilator SQL podczas przetwarzania zapytania.



Rysunek 6. Schemat blokowy analizy zapytania przez kompilator SQL

Optymalizator zapytań generuje plany dostępu lokalnego i zdalnego dla fragmentu zapytania na podstawie kosztu zasobów. Następnie program DB2 UDB wybiera plan dostępu, dla którego koszt wykonania zapytania jest najniższy.

Jeśli pewne fragmenty mają być wykonywane w źródle danych, ich przetwarzanie jest przekazywane do realizacji zdalnej. Po przetworzeniu fragmentów przez źródło danych wyniki są pobierane i zwracane do programu DB2 UDB. Jeśli część przetwarzania była realizowana przez program DB2 UDB, wyniki uzyskane lokalnie są łączone z wynikami pobranymi ze źródła danych. W końcowym etapie program DB2 UDB zwraca wszystkie wyniki klientowi.

Podstawowym celem analizy przekazywania do źródła jest określenie, które operacje mogą być wykonane zdalnie. Analizę tę przeprowadza się na podstawie otrzymanych instrukcji SQL i możliwości zdalnego źródła danych. Jej wyniki służą optymalizatorowi zapytań do oceny alternatywnych możliwości i wyboru planu dostępu według kosztów. Optymalizator może zdecydować o zaniechaniu wykonania bezpośrednio w zdalnym źródle, jeśli okaże się to nieekonomiczne. Drugorzędnym zadaniem analizy przekazywania jest próba nadania zapytaniu postaci, która lepiej nadaje się do optymalizacji przez optymalizator programu DB2 i przez zdalne optymalizatory zapytań.

Ostateczny plan dostępu wybrany przez optymalizator może składać się z operacji w zdalnym źródle danych. Dla operacji wykonywanych w każdym źródle danych w procesie zdalnego generowania SQL tworzona jest efektywna instrukcja SQL w dialekcie języka SQL obsługiwanego w źródle danych. Procedura ta, zwana *globalną optymalizacją*, pomaga utworzyć plan, który będzie optymalny dla wszystkich źródeł danych.

Dla źródeł nierelacyjnych opakowania używają protokołu żądanie-odpowiedź-kompensacja.

Pojęcia pokrewne:

- “Analiza przekazywania do źródła” na stronie 135

Zadania pokrewne:

- “Request-reply-compensate protocol” w podręczniku *IBM DB2 Information Integrator Wrapper Developer’s Guide*
- “Globalna optymalizacja” na stronie 146

Analiza przekazywania do źródła

Analiza przekazywania do źródła jest przeprowadzana w odniesieniu do relacyjnych źródeł danych. Źródła nierelacyjne używają protokołu żądanie-odpowiedź-kompensacja. Jej wyniki informują optymalizator zapytań, czy zdalne źródło danych może wykonać określoną operację. *Operacją* może być funkcja, na przykład operator relacyjny, funkcja systemowa lub użytkownika albo operator SQL (GROUP BY, ORDER BY i inne).

Funkcje, których nie można przekazać, mogą w znacznym stopniu wpłynąć na wydajność zapytania. Rozważmy efekt wymuszenia lokalnego wartościowania predykatu selekcji zamiast przekazania go do zdalnego źródła danych. Wymagałoby to pobrania przez serwer stowarzyszony całej tabeli ze zdalnego źródła danych, a następnie lokalnego przefiltrowania tej tabeli względem predykatu. Jeśli sieć jest obciążona, a tabela duża, odbije się to niekorzystnie na wydajności zapytania.

Także operatory, których nie można przekazać, w znacznym stopniu wpływają na wydajność zapytania. Na przykład, gdyby operator GROUP BY musiał lokalnie agregować zdalne dane, znowu wymagałoby to pobrania przez serwer stowarzyszony całej tabeli ze zdalnego źródła danych.

Załóżmy na przykład, że pseudonim EMP odwołuje się do tabeli EMPLOYEES. Tabela ta ma 10 000 wierszy. Jedna kolumna zawiera nazwiska pracowników, a inna wynagrodzenie

każdego pracownika. Do serwera stowarzyszonego wysyłane jest zapytanie, które ma ustalić liczbę pracowników o nazwiskach zaczynających się od litery 'B' i zarabiających ponad 5 000 zł miesięcznie.

```
SELECT LASTNAME, COUNT(*) FROM EMP
WHERE LASTNAME LIKE 'B%' AND SALARY > '50000'
GROUP BY LASTNAME;
```

Gdy kompilator SQL programu DB2[®] otrzyma taką instrukcję, przeanalizuje kilka możliwości:

- Kolejności zestawiania są takie same. Jest prawdopodobne, że predykat zapytania zostanie przekazany do źródła danych. Zazwyczaj efektywniejsze jest filtrowanie i grupowanie wyników w źródle danych zamiast kopiowania całej tabeli do serwera stowarzyszonego i wykonywanie tych operacji lokalnie. Na podstawie analizy przekazywania określa się, czy operacje mogą być wykonane w źródle danych. Ponieważ kolejności zestawiania są takie same, operacje predykatu i GROUP BY mogą być wykonane w źródle danych.
- Kolejności zestawiania są takie same i optymalizator zapytania ma informację, że serwer stowarzyszony jest bardzo szybki. Możliwe jest, że serwer stowarzyszony zdecyduje, iż lokalne wykonanie operacji GROUP BY będzie najlepszym (najtańszym) rozwiązaniem. Predykat zostanie przekazany do źródła danych w celu wartościowania. Jest to przykład analizy przekazywania połączonej z globalną optymalizacją.
- Kolejności zestawiania nie są takie same. Z analizy przekazywania wynika, że nie można całej klauzuli WHERE wartościować w źródle danych. Jednak optymalizator zapytania może zdecydować, że efektywniej będzie przekazać do wartościowania część LIKE z predykatu. Porównanie zakresu musi być jednak wykonane w stowarzyszonej bazie danych. To kolejny przykład analizy przekazywania połączonej z globalną optymalizacją.

Kompilator SQL porówna dostępne plany dostępu i wybierze najbardziej efektywny z nich.

Ogólnym celem jest zapewnienie, aby optymalizator rozważał możliwość przekazywania funkcji i operatorów do wartościowania w źródłach danych. Na to, czy dana funkcja lub operator SQL zostaną obliczone w zdalnym źródle danych, ma wpływ wiele czynników. Podstawowe czynniki, od których zależą decyzje optymalizatora zapytań, to: cechy serwera, cechy pseudonimu i cechy zapytania.

Pojęcia pokrewne:

- “Cechy serwera wpływające na możliwości przekazywania” na stronie 136
- “Cechy pseudonimów wpływające na możliwości przekazywania” na stronie 140
- “Cechy zapytań wpływające na możliwości przekazywania” na stronie 142

Zadania pokrewne:

- “Request-reply-compensate protocol” w podręczniku *IBM DB2 Information Integrator Wrapper Developer's Guide*

Analiza przekazywania do źródła - informacje szczegółowe

Cechy serwera wpływające na możliwości przekazywania

W kolejnych sekcjach opisano skojarzone ze źródłem danych czynniki, które mogą wpływać na możliwości przekazywania przetwarzania do źródła.

W przypadku relacyjnych źródeł danych istnienie tych czynników wynika z faktu, że do wprowadzania zapytań używany jest dialekt języka SQL programu DB2[®], który może oferować więcej możliwości niż dialekt SQL w źródle danych. Serwer stowarzyszony DB2

może kompensować brak określonych funkcji na serwerze danych, może to jednak wymagać wykonania operacji na serwerze stowarzyszonym.

Czynniki, od których zależą możliwości przekazywania do nierelacyjnych źródeł danych, są inne niż w przypadku źródeł relacyjnych. Dla większości nierelacyjnych źródeł danych czynnikiem takim nie jest na przykład dialekt SQL, ponieważ źródła te nie korzystają z języka SQL.

Różnice SQL

- **Możliwości języka SQL.** Poszczególne źródła danych obsługują różne odmiany dialektu SQL i oferują różne poziomy funkcjonalności. Można to zilustrować przykładem operatora GROUP BY. Większość źródeł danych obsługuje ten operator. Jednak niektóre z nich nakładają ograniczenia dotyczące liczby pozycji na liście GROUP BY lub nie zezwalają na użycie na tej liście wyrażeń. W wypadku takiego ograniczenia w zdalnym źródle danych konieczne może być wykonanie operacji GROUP BY lokalnie przez serwer stowarzyszony.
- **Zastrzeżenia dotyczące języka SQL.** W każdym źródle danych mogą występować różne zastrzeżenia języka SQL. Na przykład niektóre źródła danych wymagają znaczników parametrów, aby powiązać wartości ze zdalnymi instrukcjami SQL. Dlatego konieczne jest sprawdzenie zastrzeżeń dotyczących znaczników parametrów, aby uzyskać pewność, że każde źródło danych obsługuje taki mechanizm wiązania. Jeśli serwer stowarzyszony nie może określić dobrej metody wiązania wartości dla funkcji, funkcja ta musi być wartościowana lokalnie.
- **Ograniczenia języka SQL.** Serwer stowarzyszony może zezwalać na korzystanie z większych liczb całkowitych niż zdalne źródła danych. Jednak wartości przekraczające limity nie mogą być osadzone w instrukcjach wysyłanych do źródeł danych. Dlatego wartość funkcji lub operatora działającego na takiej stałej musi być obliczona lokalnie.
- **Specyfika serwera.** Do tej kategorii należy kilka czynników. Jednym z przykładów może być traktowanie wartości NULL podczas sortowania (na początku lub na końcu, zależnie od uporządkowania). Jeśli, na przykład, wartości NULL są sortowane w źródle danych w inny sposób niż na serwerze stowarzyszonym, nie można zdalnie wykonywać operacji ORDER BY dla wyrażeń, których wynikiem mogą być wartości puste.

Kolejność zestawiania

Nadanie opcji serwera COLLATING_SEQUENCE wartości "Y" stanowi informację dla stowarzyszonej bazy danych, że kolejność zestawiania w źródle danych jest zgodna z kolejnością zestawiania programu DB2. Ustawienie to umożliwi optymalizatorowi zapytań rozważenie przetwarzania zależnego od kolejności w źródle danych, co może poprawić wydajność.

Jeśli kolejność zestawiania w źródle danych nie jest taka sama, jak w stowarzyszonej bazie danych, otrzymane wyniki mogą być błędne. Jeśli na przykład plan zakłada użycie łączy scalających, optymalizator w maksymalnym możliwym zakresie przekaże operacje porządkowania do źródeł danych. W wypadku zastosowania różnych kolejności zestawiania tabela wynikowa łączy może być niepoprawna. Jeśli nie wiadomo, czy kolejności zestawiania w źródle danych i w programie DB2 są identyczne, opcji serwera COLLATING_SEQUENCE należy nadać wartość "N".

Rozwiązaniem alternatywnym może być skonfigurowanie w stowarzyszonej bazie danych takiej samej kolejności zestawiania, jak w źródle danych. Następnie można nadać opcji serwera COLLATING_SEQUENCE wartość 'Y'. Pozwoli to uwzględnić przez optymalizator zapytań możliwość "przekazania" znakowych predykatów zakresu.

W celu określenia, czy kolejność zestawiania w źródle danych jest taka sama, jak w programie DB2, należy rozważyć następujące czynniki:

- Obsługa języków narodowych
Kolejność zestawiania jest związana z językiem obsługiwany na serwerze. Należy porównać ustawienia narodowe systemu operacyjnego, w którym działa program DB2, z ustawieniami narodowymi systemu operacyjnego w źródle danych.
- Cechy źródła danych
Niektóre źródła danych są tworzone z zastosowaniem kolejności zestawiania, w której nie jest rozróżniana wielkość liter, co może prowadzić do innych niż w programie DB2 UDB wyników operacji zależnych od uporządkowania.
- Dostosowanie
W niektórych źródłach dostępnych jest wiele opcji określających kolejność zestawiania lub możliwe jest dostosowanie kolejności zestawiania.

Jeśli zapytanie z serwera stowarzyszonego wymaga sortowania, miejsce wykonania tej operacji zależy od kilku czynników. Jeśli kolejność zestawiania w stowarzyszonej bazie danych jest taka sama, jak w źródle danych, sortowanie może odbyć się w źródle danych. W takiej sytuacji optymalizator może zdecydować, co jest bardziej uzasadnione ekonomicznie — sortowanie lokalne, czy sortowanie zdalne. Podobnie, gdy zapytanie wymaga porównania danych znakowych, operacja ta może być również wykonana w źródle danych.

Porównania liczbowe mogą być w zasadzie wykonywane w dowolnym miejscu, nawet jeśli kolejności zestawiania są różne. Jeśli jednak waga znaków pustych w stowarzyszonej bazie danych jest inna niż w źródle danych, otrzymane wyniki mogą być niepoprawne.

Także w przypadku instrukcji porównania należy zachować ostrożność podczas wprowadzania instrukcji do źródła danych, w którym nie jest rozróżniana wielkość liter. Wagi przypisywane znakom "A" i "a" w źródle danych, które nie rozróżnia wielkości liter, są takie same. Na przykład w źródle danych nie rozróżniającym wielkości liter z angielską stroną kodową ciągi znaków STEWART, SteWArT i stewart będą uważane za równe. W stowarzyszonej bazie danych DB2 domyślnie rozróżniana jest wielkość liter i ciągi te miałyby różne wagi.

Jeśli kolejności zestawiania w stowarzyszonej bazie danych i w źródle danych różnią się, serwer stowarzyszony pobiera dane do stowarzyszonej bazy danych, dzięki czemu operacje sortowania i porównywania mogą być wykonane lokalnie. Lokalne porządkowanie danych przez serwer stowarzyszony pozwala spełnić oczekiwania użytkowników, którzy spodziewają się, że wyniki zapytania zostaną uporządkowane zgodnie z kolejnością zestawiania zdefiniowaną dla serwera stowarzyszonego.

Jeśli w zapytaniu występuje znak równości, można przekazać dany fragment także wtedy, gdy kolejności zestawiania są różne (opcja serwera ma wartość 'N'). Na przykład predykat C1 = 'A' może być przekazany do przetworzenia w źródle danych. Zapytań takich nie można oczywiście przekazywać, gdy kolejność zestawiania w źródle danych nie uwzględnia wielkości liter. Gdy w źródle danych nie jest rozróżniana wielkość liter, wartości wyrażeń C1 = 'A' i C1 = 'a' są takie same, co jest nie do przyjęcia w środowiskach, w których wielkość liter jest rozróżniana (DB2).

Administratorzy mogą tworzyć stowarzyszone bazy danych z określoną kolejnością zestawiania, która będzie zgodna z kolejnością zestawiania w źródle danych. Takie podejście może poprawić wydajność, gdy we wszystkich źródłach danych kolejność zestawiania jest taka sama, lub gdy większość funkcji działających na kolumnach jest używana w odniesieniu do źródeł danych z taką samą kolejnością zestawiania.

Pobieranie danych w celu lokalnego sortowania i porównywania zwykle wiąże się ze spadkiem wydajności. Dlatego należy rozważyć skonfigurowanie w stowarzyszonej bazie danych takiej samej kolejności zestawiania, jak w źródle danych. Może to przyczynić się do

poprawy wydajności, ponieważ serwer stowarzyszony umożliwi wykonywanie operacji sortowania i porównywania w źródłach danych. Na przykład w programie DB2 for z/OS™ and OS/390® operacje sortowania za pomocą klauzuli ORDER BY wykorzystują kolejność zestawiania opartą na stronie kodowej EBCDIC. Aby serwer stowarzyszony pobierał ze źródła DB2 for z/OS and OS/390 dane posortowane zgodnie z klauzulą ORDER BY, zaleca się skonfigurowanie w stowarzyszonej bazie danych predefiniowanej kolejności zestawiania opartej na stronie kodowej EBCDIC.

Jeśli kolejności zestawiania stowarzyszonej bazy danych i źródła danych są różne, a tabela wynikowa ma być uporządkowana w kolejności źródła danych, można wprowadzić zapytanie w sesji tranzytowej lub zdefiniować je w widoku źródła danych.

Opcje serwera stowarzyszonego

Wymienione wcześniej czynniki mające wpływ na możliwości przekazywania to cechy serwerów baz danych, których użytkownik nie może zmienić. Wymienione poniżej opcje serwera mogą być ustawiane przez użytkownika i, w niektórych przypadkach, mogą mieć wpływ na wydajność zapytań:

- **COLLATING_SEQUENCE**. Jeśli kolejność zestawiania w źródle danych różni się od kolejności zestawiania w programie DB2 dla systemów Linux, UNIX,® i Windows®, żadne operacje zależne od kolejności zestawiania programu DB2 nie mogą być wykonane zdalnie w źródle danych. Przykładem może być wykonanie funkcji kolumnowej MAX dla kolumny znakowej pseudonimu w źródle danych o innej kolejności zestawiania. Ponieważ wykonanie funkcji MAX w zdalnym źródle danych może dać inne wyniki, stowarzyszona baza danych wykona operację agregacji i wyliczy wartość funkcji MAX lokalnie.
- **VARCHAR_NO_TRAILING_BLANKS**. Ta opcja dotyczy łańcuchów znaków o zmiennej długości, które nie zawierają końcowych znaków odstępu. Niektóre źródła danych, jak Oracle, nie mają semantyki zapewniającej uzyskanie takich samych wyników porównywania danych znakowych dopełnianych spacjami, jak w programie DB2 dla systemów Linux, UNIX i Windows. Ta różnica w traktowaniu dopełniania może powodować, że wyniki będą odbiegać od oczekiwanych. Jeśli wiadomo, że kolumny typów VARCHAR i VARCHAR2 w źródle danych nie zawierają końcowych znaków odstępu, można włączyć tę opcję serwera dla źródła danych. Należy sprawdzić, czy uwzględniono wszystkie obiekty, które potencjalnie mogą mieć pseudonimy, w tym również widoki.
- **DB2_MAXIMAL_PUSHDOWN**. Ta opcja określa podstawowe kryteria, na podstawie których optymalizator zapytań wybiera plan dostępu. Optymalizator zapytań może dobrać plan dostępu na podstawie kosztu lub postawionego przez użytkownika warunku, aby jak największa część obciążenia przetwarzaniem przypadła na zdalne źródło danych. Nadanie opcji DB2_MAXIMAL_PUSHDOWN wartości 'Y' spowoduje, że podstawowym kryterium dla optymalizatora zapytań stanie się zmniejszenie ruchu w sieci. Optymalizator wybierze plan dostępu, który przewiduje najmniejszą liczbę "wysyłek" do źródła danych. Nadanie tej opcji wartości 'Y' wymusi użycie przez serwer stowarzyszony planu dostępu, dla którego koszty nie muszą być najniższe. Użycie planu dostępu innego, niż zapewniający najniższy koszt, może obniżyć wydajność. Jeśli część lub całość przetwarzania zapytania można zrealizować w oparciu o zmaterializowaną tabelę zapytania na serwerze stowarzyszonym, możliwe jest zastosowanie planu dostępu, który zakłada wykorzystanie takiej tabeli. Użycie zmaterializowanej tabeli zapytania zamiast przekazywania operacji do źródła danych zmniejsza ruch w sieci. Gdy opcja serwera DB2_MAXIMAL_PUSHDOWN ma wartość 'Y', zapytanie, którego wynik jest iloczynem kartezyjańskim, nie będzie przekazywane do zdalnego źródła danych. Zapytania takie będą przetwarzane przez stowarzyszoną bazę danych. Opcja serwera DB2_MAXIMAL_PUSHDOWN nie musi mieć wartości 'Y', aby serwer stowarzyszony przekazywał zapytania do przetwarzania w zdalnych źródłach danych. Optymalizator zapytań będzie przekazywał przetwarzanie zapytań także wtedy, gdy opcja ta będzie miała wartość 'N' (wartość domyślna). Jednak wówczas podstawowym kryterium optymalizatora będzie koszt, a nie ruch w sieci.

Uwagi dotyczące odwzorowań funkcji i typów

Domyślne odwzorowania typów danych są wbudowane w opakowania źródeł danych. Odwzorowania te są zaprojektowane w taki sposób, aby każdemu typowi danych źródła danych przydzielana była wystarczająca ilość miejsca w buforze, pozwalająca uniknąć przepełnienia buforu podczas wykonywania. Odwzorowania dla poszczególnych źródeł danych można modyfikować, dostosowując je do konkretnych aplikacji. Na przykład typ DATE kolumny źródła danych Oracle zostanie domyślnie odwzorowany na typ TIMESTAMP programu DB2 dla systemów Linux, UNIX, i Windows. Lokalny typ danych można jednak zmienić na typ danych DATE programu DB2 dla systemów Linux, UNIX i Windows. Wyeliminuje to konieczność użycia funkcji SCALAR w celu wyodrębnienia podzbioru wszystkich danych przechowywanych w typie danych TIMESTAMP.

Także domyślne odwzorowania funkcji są wbudowane w opakowania źródeł danych. Stowarzyszona baza danych kompensuje brak obsługi funkcji w źródłach danych. Kompensacja taka ma miejsce w trzech wypadkach:

- Brak funkcji w źródle danych. Na przykład w źródłach danych DB2 for z/OS and OS/390 nie są dostępne niektóre funkcje SYSFUN i brak ten musi być kompensowany lokalnie.
- Funkcja istnieje w źródle danych, ale cechy operandu naruszają ograniczenia obowiązujące dla funkcji. Przykładem może być operator relacyjny IS NULL. Jest on obsługiwany przez większość źródeł danych, ale niektóre z nich nakładają ograniczenia, na przykład dopuszczając umieszczanie nazwy kolumny wyłącznie po lewej stronie operatora IS NULL.
- Funkcja obliczana zdalnie może zwracać inną wartość. Przykładem może być operator '>' (większe niż). Dla źródeł o innych kolejnościach zestawiania wartość wyrażenia z tym operatorem może być inna niż wyliczona lokalnie w programie DB2 dla systemów Linux, UNIX i Windows.

Pojęcia pokrewne:

- “Collating Sequences” w podręczniku *Application Development Guide: Programming Client Applications*
- “Odwzorowania typów danych w systemie stowarzyszonym” na stronie 47
- “Strojenie przetwarzania zapytań” na stronie 133
- “Analiza przekazywania do źródła” na stronie 135
- “Cechy pseudonimów wpływające na możliwości przekazywania” na stronie 140
- “Cechy zapytań wpływające na możliwości przekazywania” na stronie 142

Informacje pokrewne:

- Rozdział 21, “Opcje serwera w systemie stowarzyszonym”, na stronie 225

Cechy pseudonimów wpływające na możliwości przekazywania

Istnieje kilka czynników powiązanych z pseudonimami, które mogą wpływać na możliwości przekazywania. Lokalny typ danych kolumny pseudonimu może wpłynąć na liczbę możliwych kolejności łączenia określonych przez optymalizator. Ponadto pseudonimy mogą mieć ustawioną opcję kolumny wskazującą, że dane w kolumnach nie zawierają końcowych znaków odstępu. Pozwala to kompilatorowi SQL wygenerować efektywniejszą postać predykatu dla instrukcji SQL wysyłanej do źródeł danych.

Lokalny typ danych kolumny pseudonimu

Lokalny typ danych kolumny nie powinien wykluczać możliwości wartościowania predykatu w źródle danych. Domyślne odwzorowania typów danych mają na celu uniknięcie potencjalnego przepełnienia buforu. Jednak w wypadku łączenia dwóch kolumn zawierających dane o różnej długości wartościowanie predykatu łączenia w źródle danych,

którego kolumna zawiera krótsze dane, może nie być brane pod uwagę, zależnie od sposobu wiązania kolumny z dłuższymi danymi w programie DB2[®] UDB. Ta sytuacja może mieć wpływ na liczbę możliwych kolejności łączenia określanych przez optymalizator. Na przykład kolumnom źródła danych Oracle utworzonym przy użyciu typu danych INTEGER lub INT przypisywany jest typ NUMBER(38). Kolumnie pseudonimu dla tego typu danych Oracle zostanie przypisany lokalny typ danych FLOAT, ponieważ zakres liczb całkowitych programu DB2 wynosi od $2^{*}31$ do $(-2^{*}31)-1$, co mniej więcej odpowiada typowi NUMBER(9). W takim wypadku łączenie kolumny typu integer źródła DB2 z kolumną typu integer źródła Oracle nie może odbyć się w źródle danych DB2 (krótszy typ kolumny łączonej). Jeśli jednak dziedzina kolumny Oracle pokrywa się z typem danych INTEGER programu DB2, należy zmienić lokalny typ danych tej kolumny za pomocą instrukcji ALTER NICKNAME, co pozwoli wykonać operację łączenia w źródle danych DB2.

Opcje kolumn w systemach stowarzyszonych

Opcje kolumn stanowią dla opakowania informację o konieczności traktowania danych w kolumnie w szczególny sposób. Kierując się informacjami zawartymi w metadanych, kompilator SQL i optymalizator zapytań opracowują doskonalsze plany dostępu do danych. Program DB2 UDB traktuje obiekt, do którego odwołuje się pseudonim, jak tabelę. Pozwala to ustawić opcje kolumny dla każdego obiektu źródła danych, dla którego zdefiniowano pseudonim. Do dodawania i modyfikowania opcji kolumn dla pseudonimów można używać instrukcji ALTER NICKNAME. Dostępne są dwie opcje kolumny:

- **NUMERIC_STRING**. Ta opcja dotyczy kolumn typu znakowego (CHAR i VARCHAR). Założmy, że w źródle danych ustawiona jest inna kolejność zestawiania niż w stowarzyszonej bazie danych. Serwer stowarzyszony w takiej sytuacji z reguły nie przekazywałby wykonania operacji sortowania kolumn z danymi znakowymi do źródła danych. Dane byłyby przekazywane do stowarzyszonej bazy danych i sortowane lokalnie. Założmy jednak, że kolumna ma przypisany typ znakowy, mimo że zawiera wyłącznie dane złożone z cyfr ('0', '1', ..., '9'). Informację o tym można przekazać, ustawiając opcję kolumny NUMERIC_STRING na 'Y'. Dzięki temu optymalizator zapytań DB2 uzyskuje możliwość realizacji sortowania w źródle danych. Wykonując sortowanie zdalnie, można uniknąć narzutu związanego z przekazywaniem danych na serwer stowarzyszony i wykonaniem sortowania lokalnie.
- **VARCHAR_NO_TRAILING_BLANKS**. W przeciwieństwie do opcji serwera o tej samej nazwie, ta opcja kolumny umożliwia identyfikację tych kolumn w źródle Oracle, które nie zawierają końcowych znaków odstępu. Kompilator SQL uwzględni te informacje na etapie analizy przekazywania podczas sprawdzania wszystkich operacji wykonywanych na kolumnach, dla których ta opcja jest włączona. Na podstawie wartości opcji VARCHAR_NO_TRAILING_BLANKS kompilator SQL może wygenerować różne, choć równoważne, formy predykatu używanego w zdalnej instrukcji SQL wysyłanej do źródła danych. Może się zdarzyć, że predykaty wyliczane dla źródła danych będą różne, ale wyniki końcowe powinny być równoważne.

Zmaterializowane tabele zapytań

Zmaterializowana tabela zapytania to tabela podsumowania utworzona z tabeli wynikowej zapytania. Zapytanie używane do utworzenia takiej tabeli zawiera pełną selekcję z klauzulą GROUP BY, która agreguje dane z tabeli występującej w zapytaniu.

W przeciwieństwie do widoku, zmaterializowana tabela zapytania zawiera rzeczywiste dane z tabeli wynikowej. W systemie stowarzyszonym można tworzyć zmaterializowane tabele zapytań, które odwołują się do jednego lub kilku pseudonimów, albo do kombinacji pseudonimów i tabel lokalnych.

Zmaterializowana tabela zapytania zdefiniowana na podstawie zapytania odwołującego się do pseudonimu jest lokalną tabelą na serwerze stowarzyszonym. Zawiera ona kopię danych przechowywanych w zdalnych źródłach danych, które zostały zwrócone w tabeli wynikowej zapytania. Użytkownicy systemu stowarzyszonego mogą nie wiedzieć o istnieniu

zmaterializowanej tabeli zapytania. Kiedy do serwera stowarzyszonego zostanie wysłane zapytanie odwołujące się do pseudonimu, dla którego utworzono zmaterializowaną tabelę zapytania, optymalizator zapytań może w niezauważalny sposób użyć lokalnej zmaterializowanej tabeli zapytania zamiast łączyć się ze zdalnym źródłem danych. Korzystanie z danych przechowywanych lokalnie zwiększa wydajność.

Zmaterializowane tabele zapytań można tworzyć wyłącznie z opcją REFRESH DEFERRED.

Pojęcia pokrewne:

- “Tabele buforowane” na stronie 179
- “Strojenie przetwarzania zapytań” na stronie 133
- “Analiza przekazywania do źródła” na stronie 135
- “Cechy serwera wpływające na możliwości przekazywania” na stronie 136
- “Cechy zapytań wpływające na możliwości przekazywania” na stronie 142

Zadania pokrewne:

- “Modyfikowanie pseudonimu” na stronie 34

Informacje pokrewne:

- Rozdział 24, “Opcje kolumn pseudonimu dla systemów stowarzyszonych”, na stronie 253

Cechy zapytań wpływające na możliwości przekazywania

W zapytaniu może być użyty operator SQL, który odwołuje się do pseudonimów z wielu źródeł danych. Kiedy serwer stowarzyszony łączy wyniki z dwóch źródeł danych za pomocą jednego operatora, operacja ta musi odbywać się na serwerze stowarzyszonym. Przykładem może być operator mnogościowy UNION. Operatora tego nie można wartościować bezpośrednio w zdalnym źródle danych.

Pojęcia pokrewne:

- “Cechy serwera wpływające na możliwości przekazywania” na stronie 136
- “Cechy pseudonimów wpływające na możliwości przekazywania” na stronie 140

Decyzje wynikające z analizy przekazywania do źródła

Zmodyfikowanie instrukcji SQL może stworzyć nowe możliwości przekazywania przetwarzania zapytań do źródła przez serwer stowarzyszony. W poniższych sekcjach opisano kilka narzędzi, które można wykorzystać do określenia, gdzie zapytanie jest oceniane pod kątem przekazywania do źródła, oraz przedstawiono listę typowych pytań (i sugerowanych obszarów badań) powiązanych z analizą zapytań. Omówiono także zagadnienia dotyczące aktualizacji źródła danych.

Gdzie będzie wartościowane zapytanie?

Szczegółowe informacje optymalizatora zapytań są przechowywane w tabelach wyjaśniania, niezależnie od samego planu dostępu. Informacje te umożliwiają dogłębną analizę planu dostępu. Tabele wyjaśniania są dostępne we wszystkich obsługiwanych systemach operacyjnych i zawierają informacje dotyczące instrukcji zarówno statycznego, jak i dynamicznego SQL. Dostęp do tabel wyjaśniania można uzyskać przy użyciu instrukcji SQL. Umożliwia to łatwe manipulowanie danymi wyjściowymi, porównywanie różnych zapytań lub analizowanie zmian w działaniu tego samego zapytania.

Procedura:

Istnieją dwa sposoby uzyskiwania informacji dotyczących globalnego planu dostępu z tabel wyjaśniania.

- Narzędzie do formatowania tabeli wyjaśniania. Za pomocą narzędzia **db2exfmt** można przedstawić informacje z tabel wyjaśniania w predefiniowanym formacie.
- Narzędzia **db2expln** i **dynexpln** pozwalają zapoznać się z planem dostępu wybranym dla określonej instrukcji SQL. W tym celu można także użyć zintegrowanego narzędzia Explain Facility z Centrum sterowania DB2® w połączeniu z narzędziem Visual Explain. Narzędzie Explain Facility służy do wyjaśniania instrukcji zarówno statycznego, jak i dynamicznego SQL. Różnica między nim a innymi narzędziami polega na tym, że w połączeniu z narzędziem Visual Explain informacje z programu Explain Facility są prezentowane w formacie graficznym. Poza tym poziom szczegółowości w obydwu metodach jest równoważny. Do pełnego wykorzystania danych wyjściowych programów **db2expln** i **dynexpln** wymagana jest znajomość:
 - różnych obsługiwanych instrukcji SQL i związanej z nimi terminologii (na przykład predykatów w instrukcji SELECT),
 - przeznaczenia pakietu (planu dostępu),
 - przeznaczenia i zawartości tabel katalogu systemowego,
 - ogólnych pojęć dotyczących strojenia aplikacji.

Wskazane jest użycie narzędzi wyjaśniających dla różnych wartości opcji serwera DB2_MAXIMAL_PUSHDOWN. W tym celu należy uruchomić narzędzia wyjaśniające dla zapytania, dla którego opcja serwera DB2_MAXIMAL_PUSHDOWN ma wartość 'N'. Jest to domyślne ustawienie tej opcji. Analiza przekazywania do źródła pozwala określić, które części kodu SQL mogą być przekazane. Następnie optymalizator zapytań generuje wszystkie alternatywne plany, które nie naruszają kryteriów ustalonych przez analizę przekazywania. Optymalizator zapytań szacuje koszt każdego planu i wybiera plan o najniższym prognozowanym koszcie. Następnie należy nadać opcji serwera DB2_MAXIMAL_PUSHDOWN wartość 'Y'. Ponownie należy uruchomić narzędzia wyjaśniające dla tej samej instrukcji SQL. Plan przedstawiony w wynikach programu zawierać będzie wszystkie operacje SQL, które mogą być przekazane do źródła danych. Ewentualne różnice między dwoma planami będą wynikać z informacji użytych przez optymalizator zapytań do optymalizacji kosztów przetwarzania. Informacje te dotyczą indeksu pseudonimu, statystyk pseudonimu i atrybutów serwera.

Pojęcia pokrewne:

- “Explain tools” w podręczniku *Administration Guide: Performance*
- “SQL explain tools” w podręczniku *Administration Guide: Performance*
- “dynexpln” w podręczniku *Administration Guide: Performance*
- “Description of db2expln and dynexpln output” w podręczniku *Administration Guide: Performance*
- “Strojenie przetwarzania zapytań” na stronie 133
- “Analiza przekazywania do źródła” na stronie 135
- “Przebieg oceny planu dostępu” na stronie 144
- “Uaktualnianie i dostosowywanie źródła danych” na stronie 146

Zadania pokrewne:

- “Globalna optymalizacja” na stronie 146

Informacje pokrewne:

- “db2exfmt - Explain Table Format Command” w podręczniku *Command Reference*

- “db2expln - SQL Explain Command” w podręczniku *Command Reference*

Przebieg oceny planu dostępu

W tej sekcji przedstawiono typowe pytania dotyczące analizy planu dostępu oraz obszary, które można zbadać, aby zwiększyć możliwości przekazywania.

Dlaczego dany predykat nie jest wartościowany zdalnie?

Pytanie to powstaje, gdy predykat ma charakter bardzo selektywny, dzięki czemu mógłby być użyty do przefiltrowania wierszy i zmniejszenia ruchu w sieci. Zdalne wyliczanie predykatu ma także wpływ na to, czy łączenie dwóch tabel z tego samego źródła danych może być wykonane zdalnie.

Obszary, które należy zbadać, to:

- Predykaty podzapytań. Czy dany predykat zawiera podzapytanie, które dotyczy innego źródła danych? Czy dane zapytanie zawiera podzapytanie z operatorem SQL, który nie jest obsługiwany przez to źródło danych? Nie wszystkie źródła danych obsługują operatory mnogościowe w predykacie.
- Funkcje predykatów. Czy dany predykat zawiera funkcję, która nie może być wyliczona w tym źródle danych? Do funkcji zalicza się operatory relacyjne.
- Wymagania predykatu w zakresie wiązania. Czy zdalne wartościowanie danego predykatu wymaga powiązania z nim jakiejś wartości? Jeśli tak, czy naruszyłoby to ograniczenia SQL w tym źródle danych?
- Optymalizacja globalna. Optymalizator może zdecydować, że lokalne przetwarzanie jest bardziej ekonomiczne.

Dlaczego operator GROUP BY nie jest wartościowany zdalnie?

Istnieje kilka obszarów, które można zbadać:

- Czy dane wejściowe operatora GROUP BY są wyliczane zdalnie? Jeśli nie, należy sprawdzić te dane.
- Czy w źródle danych obowiązują jakieś ograniczenia dotyczące tego operatora? Przykłady takich ograniczeń to:
 - ograniczona liczba pozycji operatora GROUP BY,
 - ograniczona łączna wielkość (w bajtach) wszystkich pozycji operatora GROUP BY,
 - tylko specyfikacje kolumn na liście operatora GROUP BY.
- Czy źródło danych obsługuje ten operator SQL?
- Optymalizacja globalna. Optymalizator może zdecydować, że lokalne przetwarzanie jest bardziej ekonomiczne.

Dlaczego operator SET nie jest wartościowany zdalnie?

Istnieje kilka obszarów, które można zbadać:

- Czy obydwa operandy są w całości w tym samym zdalnym źródle danych? Jeśli tak nie jest, a być powinno, należy sprawdzić każdy operand.
- Czy w źródle danych obowiązują jakiegokolwiek ograniczenia dotyczące tego operatora SET? Czy na przykład duże obiekty lub pola długie są prawidłowymi danymi wejściowymi dla tego operatora SET?

Dlaczego operacja ORDER BY nie jest wartościowana zdalnie?

Należy wziąć pod uwagę:

- Czy dane wejściowe operacji ORDER BY są wyliczane zdalnie? Jeśli nie, należy sprawdzić te dane.

- Czy klauzula ORDER BY zawiera wyrażenie znakowe? Jeśli tak, to czy kolejność zestawiania w zdalnym źródle danych jest inna niż na serwerze stowarzyszonym?
- Czy w źródle danych obowiązują jakieś ograniczenia dotyczące tego operatora? Na przykład, czy liczba pozycji operatora ORDER BY jest ograniczona? Czy źródło danych ogranicza specyfikację kolumny dla listy ORDER BY?

Dlaczego zdalna instrukcja INSERT z pełną selekcją nie jest w całości wartościowana zdalnie?

Należy wziąć pod uwagę:

- Czy podselekcja może być w całości wykonana w zdalnym źródle danych? Jeśli nie, należy sprawdzić podselekcję.
- Czy podselekcja zawiera operator mnogościowy? Jeśli tak, to czy to źródło danych obsługuje operatory mnogościowe jako dane wejściowe dla instrukcji INSERT?
- Czy podselekcja odwołuje się do tabeli docelowej? Jeśli tak, to czy to źródło danych dopuszcza taką składnię?

Dlaczego zdalna instrukcja INSERT z klauzulą VALUES nie jest w całości wartościowana zdalnie?

Należy wziąć pod uwagę:

- Czy klauzula VALUES może być w całości przetworzona w zdalnym źródle danych? Innymi słowy, czy wyrażenie zawiera funkcję, która nie jest obsługiwana w zdalnym źródle danych?
- Czy wyrażenie zawiera podzapytanie skalarne? Czy taka składnia jest obsługiwana?
- Czy wyrażenie odwołuje się do tabeli docelowej? Czy taka składnia jest obsługiwana?

Dlaczego zdalna instrukcja UPDATE z wyszukiwaniem nie jest w całości wartościowana zdalnie?

Należy wziąć pod uwagę:

- Czy klauzula SET może być w całości wartościowana w zdalnym źródle danych? Innymi słowy, czy wyrażenie aktualizacji zawiera funkcję, która nie jest obsługiwana w zdalnym źródle danych?
- Czy klauzula SET zawiera podzapytanie skalarne? Czy źródło danych dopuszcza taką składnię?
- Czy warunek wyszukiwania może być w całości wartościowany w zdalnym źródle danych? Jeśli nie, należy sprawdzić warunek wyszukiwania.
- Czy warunek wyszukiwania lub klauzula SET odwołuje się do tabeli docelowej? Czy źródło danych dopuszcza taką składnię?
- Czy warunek wyszukiwania lub klauzula SET odwołuje się do tabeli docelowej z korelacją? Czy źródło danych dopuszcza taką składnię?

Dlaczego zdalna pozycjonowana instrukcja UPDATE nie jest w całości wartościowana zdalnie?

Dzieje się tak wtedy, gdy program DB2[®] UDB ustali wartość wyrażenia aktualizującego lokalnie, przed wysłaniem instrukcji UPDATE do źródła danych. Takie rozwiązanie nie powinno mieć znaczącego wpływu na wydajność.

- Czy klauzula SET może być w całości wartościowana w zdalnym źródle danych? Innymi słowy, czy wyrażenie aktualizacji zawiera funkcję, która nie jest obsługiwana w zdalnym źródle danych?
- Czy klauzula SET zawiera podzapytanie skalarne? Czy źródło danych dopuszcza taką składnię?

Dlaczego zdalna instrukcja DELETE z wyszukiwaniem nie jest w całości wartościowana zdalnie?

Należy wziąć pod uwagę:

- Czy warunek wyszukiwania może być w całości wartościowany w zdalnym źródle danych? Jeśli nie, należy sprawdzić warunek wyszukiwania.
- Czy warunek wyszukiwania odwołuje się do tabeli docelowej? Czy źródło danych dopuszcza taką składnię?
- Czy warunek wyszukiwania odwołuje się do tabeli docelowej z korelacją? Czy źródło danych dopuszcza taką składnię?

Pojęcia pokrewne:

- “Analiza przekazywania do źródła” na stronie 135
- “Gdzie będzie wartościowane zapytanie?” na stronie 142

Uaktualnianie i dostosowywanie źródła danych

Kompilator SQL programu DB2[®] wykorzystuje informacje przechowywane w katalogu globalnym do rozpoznania możliwości języka SQL zaimplementowanego w źródłach danych. Informacje te muszą być okresowo aktualizowane. Nowe wersje źródeł danych mogą oferować inne możliwości w zakresie obsługi języka SQL. Po uaktualnieniu źródeł danych lub ich dostosowaniu należy zaktualizować informacje w katalogu globalnym, aby kompilator SQL korzystał z najnowszych informacji.

Do zaktualizowania katalogu należy użyć instrukcji SQL DDL programu DB2, takich jak CREATE FUNCTION MAPPING i ALTER SERVER.

Pojęcia pokrewne:

- “Strojenie przetwarzania zapytań” na stronie 133
- “Analiza przekazywania do źródła” na stronie 135

Zadania pokrewne:

- “Globalna optymalizacja” na stronie 146

Globalna optymalizacja

Kompilator SQL działa dwufazowo, co pomaga opracować optymalną strategię dostępu w celu wartościowania zapytania odwołującego się do zdalnego źródła danych. Te dwie fazy to generowanie zdalnych instrukcji SQL i globalna optymalizacja. Dla zapytania wprowadzanego do stowarzyszonej bazy danych strategia dostępu może obejmować podział zapytania na zestaw fragmentów zapytania, a następnie połączenie otrzymanych wyników.

Posługując się wynikami otrzymanymi w fazie analizy przekazywania do źródła, optymalizator zapytań decyduje o miejscu wykonania każdej operacji. Operacja może być wykonana lokalnie na serwerze stowarzyszonym DB2 lub zdalnie w źródle danych. Decyzja zależy od danych wyjściowych wyrafinowanego modelu kosztów stałych używanego przez optymalizator. Model ten określa:

- koszt wykonania operacji,
- koszt przesyłania danych lub komunikatów między serwerem stowarzyszonym DB2 a źródłami danych.

Celem jest wygenerowanie zoptymalizowanego zapytania. Zoptymalizowane zapytanie to zapytanie z planem dostępu, który optymalizuje operacje zapytania dla wszystkich źródeł

danych, globalnie w całym systemie stowarzyszonym. Stan *globalnej optymalizacji* osiąga się, gdy zostanie wybrany plan dostępu o najniższych kosztach.

Kompilator SQL w programie DB2 dysponuje bazą wiedzy optymalizatora zapytań, która zawiera informacje dotyczące rodzimych źródeł danych. Optymalizator nie generuje zdalnych planów dostępu, które nie mogłyby być wygenerowane przez konkretne systemy zarządzania baz danych (DBMS). Innymi słowy, optymalizator unika sytuacji, w której wygenerowany przez niego plan dostępu byłby niezrozumiały dla optymalizatorów w zdalnych źródłach danych lub nie był przez nie akceptowany.

Wiele czynników ma wpływ na wyniki globalnej optymalizacji i tym samym na wydajność zapytania. Podstawowe czynniki to cechy serwerów i pseudonimów.

Relacyjne i nierelacyjne opakowania różnią się szczegółami, co do sposobu tworzenia planu dostępu, ale idea i ostateczne wyniki są takie same.

Pojęcia pokrewne:

- “Strojenie przetwarzania zapytań” na stronie 133
- “Analiza przekazywania do źródła” na stronie 135
- “Cechy serwera wpływające na globalną optymalizację” na stronie 147

Zadania pokrewne:

- “Cechy pseudonimu wpływające na globalną optymalizację” na stronie 149

Globalna optymalizacja - informacje szczegółowe

Chociaż globalna optymalizacja zależy od właściwości zarówno serwera, jak i pseudonimu, te ostatnie mają większy wpływ na szacunki kosztów generowane przez optymalizator zapytań.

Cechy serwera wpływające na globalną optymalizację

Informacje dotyczące cech serwera źródła danych przekazuje się do optymalizatora zapytań za pośrednictwem ustawień opcji serwera. Ustawienia te stanowią część definicji serwera źródła danych. Opcje serwera można ustawić w instrukcji CREATE SERVER podczas tworzenia definicji serwera. Za pomocą instrukcji ALTER SERVER można dodać opcje serwera do istniejącej definicji serwera. Ustawienia opcji serwera są przechowywane w katalogu globalnym stowarzyszonej bazy danych.

Opcje te dzielą się na opcje lokalne (na przykład nazwa komputera źródła danych), opcje ochrony (na przykład informacje uwierzytelniania) i opcje wydajności (na przykład współczynnik szybkości procesora).

Opcje wydajności pomagają optymalizatorowi określić, czy operacje wartościowania mogą być wykonywane w źródłach danych. Opcje serwera mające wpływ na wydajność, które mogą wymagać strojenia, to:

- CPU_RATIO
- IO_RATIO
- COMM_RATE
- COLLATING_SEQUENCE
- PLAN_HINTS

Podczas strojenia opcji serwera CPU_RATIO, IO_RATIO lub COMM_RATE należy zachować ostrożność, ponieważ gdy kalkulacja kosztów dla zapytania będzie powodować nadmiar lub niedomiar, mogą zostać wyświetlone nieoczekiwane komunikaty o błędach.

Względny stosunek szybkości procesorów

Ta wartość wskazuje, o ile szybszy lub wolniejszy jest procesor używany przez źródło danych w porównaniu z procesorem używanym przez program DB2®. Niska wartość informuje, że procesor stacji roboczej źródła danych jest szybszy niż procesor stacji roboczej programu DB2. Dla takich wartości optymalizator bierze pod uwagę przekazywanie operacji intensywnie wykorzystujących zasoby procesora do źródła danych. Niska wartość stosunku to wartość mniejsza niż 1.

Nadanie opcji serwera CPU_RATIO wartości 1 wskazuje, że procesory używane przez serwer stowarzyszony DB2 i serwer źródła danych są tak samo szybkie (stosunek 1:1). Jeśli procesor serwera stowarzyszonego DB2 jest o 50% wolniejszy niż procesor serwera źródła danych, wówczas wartość opcji serwera CPU_RATIO powinna wynosić 0,5. Jeśli natomiast procesor serwera stowarzyszonego DB2 jest dwa razy szybszy niż procesor serwera źródła danych, wówczas wartość opcji serwera CPU_RATIO powinna wynosić 2.

Względny stosunek szybkości podsystemu we/wy

Ta wartość wskazuje, o ile szybciej lub wolniej realizowane są operacje we/wy w źródle danych w porównaniu z serwerem stowarzyszonym. Niska wartość wskazuje, że podsystem we/wy źródła danych jest szybszy niż podsystem we/wy stacji roboczej programu DB2. Dla takich wartości optymalizator bierze pod uwagę przekazywanie operacji intensywnie wykorzystujących podsystem we/wy. Niska wartość stosunku to wartość mniejsza niż 1.

Nadanie opcji serwera IO_RATIO wartości 1 wskazuje, że podsystemy we/wy używane przez serwer stowarzyszony DB2 i serwer źródła danych są tak samo szybkie (stosunek 1:1). Jeśli podsystem we/wy serwera stowarzyszonego DB2 jest o 50% wolniejszy niż podsystem we/wy serwera źródła danych, wówczas wartość opcji serwera IO_RATIO powinna wynosić 0,5. Jeśli natomiast podsystem we/wy serwera stowarzyszonego DB2 jest dwa razy szybszy niż podsystem we/wy serwera źródła danych, wówczas wartość opcji serwera IO_RATIO powinna wynosić 2.

Szybkość komunikacji między serwerem stowarzyszonym a źródłem danych

Niska szybkość komunikacji wskazuje na wolne łącze komunikacyjne między serwerem stowarzyszonym a źródłem danych. Takie warunki powodują, że optymalizator zapytań stara się zmniejszyć liczbę komunikatów wysyłanych z i do źródła danych. Jeśli opcji serwera COMM_RATE zostanie nadana bardzo mała wartość, optymalizator utworzy zapytanie, które będzie generowało minimalny ruch w sieci.

Kolejność zestawiania w źródle danych

Wybór kolejności zestawiania może mieć wpływ na wydajność stowarzyszonej bazy danych. Opcja serwera COLLATING_SEQUENCE informuje, czy kolejności zestawiania w źródle danych i w stowarzyszonej bazie danych DB2 są zgodne. Program DB2 UDB może przekazać zależne od uporządkowania przetwarzanie danych znakowych do źródła danych. Jeśli kolejność zestawiania w źródle danych jest inna niż w stowarzyszonej bazie danych, optymalizator uważa dane pobrane z tego źródła za nieuporządkowane. W takiej sytuacji program DB2 UDB pobierze odpowiednie dane i w wykona całe przetwarzanie zależne od uporządkowania lokalnie (co może obniżyć wydajność). Zagadnienia dotyczące kolejności zestawiania omówiono w sekcji *Cechy serwera mające wpływ na możliwości przekazywania*.

Wskazówki dotyczące planu zdalnego

Do generowania zdalnych wskazówek dotyczących planu służy opcja serwera PLAN_HINTS. Wskazówki dotyczące planu są to fragmenty instrukcji, które dostarczają dodatkowych

informacji optymalizatorom źródeł danych. Informacje takie mogą poprawić wydajność niektórych rodzajów zapytań. Na podstawie wskazówek dotyczących planu optymalizator może łatwiej zdecydować, czy należy korzystać z indeksu, który to powinien być indeks, oraz jaką kolejność łączenia tabel należy zastosować.

Aby określić, czy ta opcja serwera poprawi wydajność zapytań, należy przeprowadzić testy.

Nie można umieścić własnych wskazówek dotyczących planu w zapytaniu.

Jeśli wskazówki dotyczące planu są włączone, zapytanie wysłane do źródła danych będzie zawierało dodatkowe informacje. Przykładowa instrukcja ze wskazówkami dotyczącymi planu wysłana do optymalizatora Oracle może wyglądać następująco:

```
SELECT /*+ INDEX (tabela1, indeks_t1)*/  
  kol1  
FROM tabela1
```

Wskazówka dotycząca planu w tym przykładzie to łańcuch `/*+ INDEX (tabela1, indeks_t1)*/`

Pojęcia pokrewne:

- “Strojenie przetwarzania zapytań” na stronie 133
- “Cechy serwera wpływające na możliwości przekazywania” na stronie 136

Zadania pokrewne:

- “Modyfikowanie definicji serwerów i opcji serwera” na stronie 28
- “Globalna optymalizacja” na stronie 146
- “Cechy pseudonimu wpływające na globalną optymalizację” na stronie 149

Informacje pokrewne:

- Rozdział 21, “Opcje serwera w systemie stowarzyszonym”, na stronie 225

Cechy pseudonimu wpływające na globalną optymalizację

Istnieje kilka czynników związanych z pseudonimami, które mogą wpływać na globalną optymalizację. Należą do nich między innymi informacje o indeksie i statystyka katalogu globalnego.

Duże znaczenie ma aktualność informacji o indeksie i danych statystycznych katalogu globalnego dostępnych dla kompilatora SQL.

Specyfikacje indeksu

Kompilator SQL wykorzystuje informacje o indeksie do optymalizowania zapytań. Informacje o indeksie tabeli źródła danych są pozyskiwane wyłącznie podczas tworzenia pseudonimu dla tej tabeli. Po utworzeniu pseudonimu informacje o indeksie tej tabeli źródła danych dostępne na serwerze stowarzyszonym nie są aktualizowane. Jeśli zdalny indeks zostanie zmieniony, w celu zaktualizowania informacji o indeksie przechowywanych na serwerze stowarzyszonym należy usunąć pseudonim tej tabeli i utworzyć go ponownie. Alternatywnie, w razie dodania nowego indeksu do tabeli w źródle danych można także zdefiniować specyfikację indeksu dla tej tabeli na serwerze stowarzyszonym.

Informacje o indeksach nie są zbierane dla pseudonimów obiektów, które nie mają indeksów, takich jak widoki, pseudonimy lub obiekty nierelacyjnych źródeł danych.

Dla obiektu z pseudonimem, który nie ma indeksu, można utworzyć specyfikację indeksu. Specyfikacje indeksu tworzą definicje indeksu w katalogu globalnym. Specyfikacja indeksu nie jest rzeczywistym indeksem. Specyfikację indeksu tworzy się za pomocą instrukcji CREATE INDEX z klauzulą SPECIFICATION ONLY. Składnia instrukcji tworzącej specyfikację indeksu dla pseudonimu jest podobna do składni instrukcji tworzącej indeks w lokalnej tabeli.

Utworzenie specyfikacji indeksu należy wziąć pod uwagę, gdy:

- W tabeli zostaje utworzony nowy indeks.
- Tworzy się pseudonim dla obiektu źródła danych, który nie zawiera indeksów, taki jak widok lub synonim.

Przed użyciem instrukcji CREATE INDEX...SPECIFICATION ONLY dla pseudonimu widoku ze źródła danych należy wziąć pod uwagę poniższe wskazówki:

- Jeśli zdalny widok jest prostą instrukcją SELECT dla tabeli źródła danych z indeksem, utworzenie dla pseudonimu specyfikacji indeksu zgodnej z indeksem tabeli w źródle danych może w znacznym stopniu zwiększyć wydajność zapytania.
- Utworzenie specyfikacji indeksu dla zdalnego widoku, który nie jest prostą instrukcją SELECT (na przykład widoku utworzonego w wyniku łączenia dwóch tabel) może spowodować pogorszenie wydajności zapytania.

Żałujemy, że utworzono specyfikację indeksu dla zdalnego widoku, który jest połączeniem dwóch tabel. Optymalizator może wybrać ten widok jako element wewnętrzny łączenia zagnieżdżonego. Wydajność zapytania może być niska, ponieważ łączenie będzie wartościowane kilka razy. Rozwiązaniem alternatywnym jest utworzenie pseudonimów dla każdej tabeli, do której odwołuje się widok w źródle danych i utworzenie stowarzyszonego widoku, który będzie się odwoływał do obydwu pseudonimów.

Statystyka katalogu globalnego

W celu zoptymalizowania obsługi zapytań w stowarzyszonej bazie danych dla obiektów z pseudonimami wykorzystywana jest statystyka katalogu. Statystyka ta jest pobierana ze źródła danych podczas tworzenia pseudonimu dla obiektu źródła danych przy użyciu instrukcji CREATE NICKNAME. W stowarzyszonej bazie danych sprawdzana jest obecność określonego obiektu w źródle danych, a następnie podejmowana jest próba zebrania dostępnych w źródle danych informacji statystycznych. Przydatne dla optymalizatora informacje są odczytywane z katalogów źródła danych i zapisywane w katalogu globalnym na serwerze stowarzyszonym. Ponieważ optymalizator może wykorzystywać niektóre lub wszystkie informacje pochodzące z katalogu źródła danych, przed utworzeniem pseudonimu zalecane jest zaktualizowanie statystyki w źródle danych (przez użycie w źródle danych komendy odpowiadającej komendzie RUNSTATS).

Statystyka w katalogu opisuje ogólną wielkość tabel i widoków oraz zakresy wartości w skojarzonych kolumnach. Pobierane są następujące informacje:

- liczba wierszy w obiekcie pseudonimu,
- liczba stron zajmowanych przez pseudonim,
- liczba różnych wartości w każdej kolumnie tabeli,
- liczba różnych wartości w kolumnach indeksu,
- najwyższa i najniższa wartość w kolumnie.

Chociaż stowarzyszona baza danych może pobierać dane statystyczne przechowywane w źródle danych, nie wykrywa ona automatycznie aktualizacji tych danych. Ponadto stowarzyszona baza danych nie ma mechanizmu obsługi definicji obiektów ani strukturalnych zmian obiektów w źródłach danych (kiedy na przykład do tabeli dodawana jest kolumna).

Jeśli dane statystyczne lub charakterystyki strukturalne obiektu zdalnego, dla którego zdefiniowano pseudonim, zmienia się, aktualizację statystyk można wykonać na trzy sposoby:

- Użyj narzędzia do aktualizowania statystyk pseudonimu dostępnego w Centrum sterowania DB2.
- Uruchom odpowiednik komendy RUNSTATS w źródle danych. Następnie usuń bieżący pseudonim i utwórz go ponownie. Jest to zalecana metoda aktualizacji statystyki.
- Zaktualizuj statystykę ręcznie w widoku katalogu SYSSTAT.TABLES. Ta metoda jest polecana tylko wtedy, gdy wiadomo, że informacje statystyczne w zdalnym źródle danych są niepoprawne lub niekompletne.

Aktualizowanie informacji po zmianie wierszy:

Stowarzyszona baza danych nie jest informowana o operacjach dodawania lub usuwania wierszy wykonywanych bezpośrednio w źródle danych. W takiej sytuacji można jednak zaobserwować spadek wydajności, ponieważ optymalizator podejmuje decyzje na podstawie nieaktualnych informacji. Aktualizacja statystyki dla pseudonimu umożliwi optymalizatorowi opracowywanie planów dostępu dla zapytań kierowanych do źródła danych na podstawie aktualnych danych.

Aktualizowanie informacji po zmianie kolumn:

Po dodaniu, zmianie lub usunięciu kolumn w źródle danych wyniki zapytań mogą być niepoprawne lub może zostać wyświetlony komunikat o błędzie. Niech pseudonim *EUROSALES* odwołuje się do tabeli *europe* w bazie danych Sybase. Jeśli do tej tabeli zostanie dodana nowa kolumna o nazwie *CZECH*, informacje o tym fakcie nie zostaną przekazane do stowarzyszonej bazy danych. Zapytania odwołujące się do tej kolumny spowodują wygenerowanie komunikatu o błędzie.

Aby zaktualizować statystykę obiektu źródła danych w katalogu stowarzyszonej bazy danych po zmianie kolumn w tym obiekcie, należy wykonać kilka czynności:

1. Uruchom w źródle danych odpowiednik programu narzędziowego DB2 RUNSTATS. Spowoduje to zaktualizowanie statystyki przechowywanej w katalogu źródła danych.
2. Usuń bieżący pseudonim obiektu źródła danych za pomocą instrukcji DROP NICKNAME.
3. Ponownie utwórz pseudonim za pomocą instrukcji CREATE NICKNAME.

Po wykonaniu tych czynności informacje statystyczne pseudonimu będą aktualne i spójne ze schematem obiektu źródła danych.

Pojęcia pokrewne:

- “Narzędzie do aktualizacji statystyk pseudonimu - przegląd” na stronie 191
- “Cechy pseudonimów wpływające na możliwości przekazywania” na stronie 140
- “Cechy serwera wpływające na globalną optymalizację” na stronie 147

Zadania pokrewne:

- “Usuwanie pseudonimu” na stronie 44
- “Tworzenie specyfikacji indeksów dla obiektów źródła danych” na stronie 74

Decyzje dotyczące globalnej optymalizacji

W poniższych sekcjach przedstawiono kilka narzędzi, których można użyć do analizowania optymalizacji zapytań, a także listę typowych pytań (i sugerowane obszary badań) dotyczących optymalizowania zapytań.

Globalna optymalizacja - analiza

Szczegółowe informacje optymalizatora zapytań są przechowywane w tabelach wyjaśniania, niezależnie od samego planu dostępu. Informacje te umożliwiają dogłębną analizę planu dostępu. Tabele wyjaśniania są dostępne we wszystkich obsługiwanych systemach operacyjnych i zawierają informacje dotyczące instrukcji zarówno statycznego, jak i dynamicznego SQL. Dostęp do tabel wyjaśniania można uzyskać przy użyciu instrukcji SQL. Umożliwia to łatwe manipulowanie danymi wyjściowymi, porównywanie różnych zapytań lub analizowanie zmian w działaniu tego samego zapytania.

Procedura:

Istnieją dwa sposoby uzyskiwania informacji dotyczących globalnego planu dostępu z tabel wyjaśniania.

- Narzędzie do formatowania tabeli wyjaśniania. Za pomocą narzędzia **db2exfmt** można przedstawić informacje z tabel wyjaśniania w predefiniowanym formacie.
- Narzędzia **db2expln** i **dynexpln** pozwalają zapoznać się z planem dostępu wybranym dla określonej instrukcji SQL. W tym celu można także użyć zintegrowanego narzędzia Explain Facility z Centrum sterowania DB2® w połączeniu z narzędziem Visual Explain. Narzędzie Explain Facility służy do wyjaśniania instrukcji zarówno statycznego, jak i dynamicznego SQL. Różnica między nim a innymi narzędziami polega na tym, że w połączeniu z narzędziem Visual Explain informacje z programu Explain Facility są prezentowane w formacie graficznym. Poza tym poziom szczegółowości w obydwu metodach jest równoważny. Do pełnego wykorzystania danych wyjściowych programów **db2expln** i **dynexpln** wymagana jest znajomość:
 - różnych obsługiwanych instrukcji SQL i związanej z nimi terminologii (na przykład predykatów w instrukcji SELECT),
 - przeznaczenia pakietu (planu dostępu),
 - przeznaczenia i zawartości tabel katalogu systemowego,
 - ogólnych pojęć dotyczących strojenia aplikacji.

Pojęcia pokrewne:

- “Explain tools” w podręczniku *Administration Guide: Performance*
- “SQL explain tools” w podręczniku *Administration Guide: Performance*
- “dynexpln” w podręczniku *Administration Guide: Performance*
- “Description of db2expln and dynexpln output” w podręczniku *Administration Guide: Performance*
- “Strojenie przetwarzania zapytań” na stronie 133
- “Decyzje zapadające podczas optymalizacji planu dostępu” na stronie 152

Zadania pokrewne:

- “Globalna optymalizacja” na stronie 146

Informacje pokrewne:

- “db2exfmt - Explain Table Format Command” w podręczniku *Command Reference*
- “db2expln - SQL Explain Command” w podręczniku *Command Reference*

Decyzje zapadające podczas optymalizacji planu dostępu

W tej sekcji przedstawiono typowe pytania dotyczące optymalizacji oraz obszary, które można zbadać, aby poszerzyć możliwości przekazywania przetwarzania do źródła.

Dlaczego łączenie między dwoma pseudonimami obiektów z tego samego źródła danych nie jest przeprowadzane zdalnie?

Obszary, które należy zbadać, to:

- Operatory łączenia. Czy źródło danych obsługuje takie operatory?
- Predykaty łączenia. Czy predykat łączenia może być wartościowany w zdalnym źródle danych? Jeśli nie, należy sprawdzić warunek łączenia.
- Liczba wierszy w wyniku łączenia. Liczbę wierszy można określić za pomocą programu Visual Explain. Czy łączenie tworzy dużo większy zestaw wierszy niż suma wierszy w obydwu pseudonimach? Czy ta liczba wierszy jest uzasadniona? Jeśli nie, należy zastanowić się nad aktualizacją statystyk pseudonimu za pomocą programu narzędziowego RUNSTATS.

Dlaczego operator GROUP BY nie jest wartościowany zdalnie?

Obszary, które należy zbadać, to:

- Składnia operatora. Należy sprawdzić, czy operator może być wartościowany w zdalnym źródle danych.
- Liczba wierszy. Należy sprawdzić za pomocą programu Visual Explain szacowaną liczbę wierszy na wejściu i wyjściu operatora GROUP BY. Czy te dwie liczby są prawie równe? Jeśli tak, optymalizator uzna za efektywniejsze wartościowanie tego operatora GROUP BY lokalnie. Czy te liczby są uzasadnione? Jeśli nie, należy zastanowić się nad aktualizacją statystyki pseudonimu za pomocą programu narzędziowego RUNSTATS.

Dlaczego cała instrukcja nie jest wartościowana zdalnie?

Optymalizator przeprowadza optymalizację pod kątem kosztów. Nawet jeśli analizy przekazywania do źródła wskazują, że każdy operator może być wartościowany w zdalnym źródle danych, optymalizator w dalszym ciągu opiera się na szacunkach kosztów, aby wygenerować plan optymalny globalnie. Istnieje wiele czynników, które taki plan musi uwzględniać. Przyjmijmy, że zdalne źródło danych może przetworzyć każdą operację z pierwotnego zapytania. Jednak używany przez nie procesor jest znacznie wolniejszy niż procesor serwera stowarzyszonego. Może się wtedy okazać, że korzystniejsze będzie wykonanie tych operacji przez serwer stowarzyszony DB2[®]. Jeśli wyniki nie będą satysfakcjonujące, należy sprawdzić statystykę w tabeli katalogu SYSSTAT.SERVEROPTIONS.

Dlaczego plan wygenerowany przez optymalizator i w całości wykonany zdalnie cechuje się znacznie niższą wydajnością niż bezpośrednio wykonanie pierwotnego zapytania w zdalnym źródle danych?

Obszary, które należy zbadać, to:

- Zdalna instrukcja SQL wygenerowana przez optymalizator zapytań DB2. Należy sprawdzić, czy jest ona identyczna z pierwotnym zapytaniem. Należy zwrócić uwagę na zmiany w kolejności predykatów. Dobry optymalizator zapytań nie powinien być wrażliwy na kolejność predykatów w zapytaniu. Niestety nie wszystkie optymalizatory w systemach DBMS działają w identyczny sposób. Jest prawdopodobne, że optymalizator w zdalnym źródle danych wygeneruje inny plan dostępu na podstawie wejściowej kolejności predykatów. Jeśli tak jest, to problem leży po stronie zdalnego optymalizatora. Należy rozważyć zmianę kolejności predykatów w zapytaniu wprowadzanym do programu DB2 UDB lub skontaktować się z działem wsparcia dostawcy zdalnego źródła danych w celu uzyskania pomocy.

Należy także rozważyć możliwość wymiany predykatów. Dobry optymalizator zapytań nie powinien być wrażliwy na równoważne zamienniki predykatów. Możliwe, że optymalizator w zdalnym źródle danych wygeneruje inny plan na podstawie wejściowej kolejności predykatów. Na przykład niektóre optymalizatory nie mogą generować przechodnich instrukcji domknięcia dla predykatów.

- Liczba zwracanych wierszy. Liczbę tę można uzyskać z programu Visual Explain. Jeśli zapytanie zwraca bardzo dużo wierszy, potencjalnym wąskim gardłem jest ruch w sieci.
- Funkcje dodatkowe. Czy zdalna instrukcja SQL zawiera więcej funkcji niż pierwotne zapytanie? Niektóre dodatkowe funkcje mogą być wygenerowane w celu konwersji typów danych. Należy sprawdzić, czy funkcje te są niezbędne.

Pojęcia pokrewne:

- “Analiza przekazywania do źródła” na stronie 135
- “Przebieg oceny planu dostępu” na stronie 144
- “Globalna optymalizacja - analiza” na stronie 152

Zadania pokrewne:

- “Globalna optymalizacja” na stronie 146

Elementy monitora systemu mające wpływ na wydajność

Monitor systemu baz danych DB2 zbiera informacje dotyczące bieżącego stanu menedżera bazy danych oraz informacje dotyczące aktywności, takie jak liczniki i inne miary przetwarzania w bazie danych.

Za pomocą monitora systemu baz danych DB2 można w systemie stowarzyszonym zbierać informacje dotyczące aktywności bazy danych, wydajności systemu i wydajności aplikacji.

Parametr monitora o nazwie Timestamp służy do śledzenia czasów odpowiedzi w interakcjach stowarzyszonej bazy danych ze źródłem danych. Elementy danych systemu stowarzyszonego monitorowane przy użyciu parametru Timestamp to:

- czas odpowiedzi podczas tworzenia pseudonimu,
- czas odpowiedzi podczas usuwania,
- czas odpowiedzi podczas wstawiania,
- czas tranzytu,
- czas odpowiedzi zapytania,
- czas uzyskania zdalnej blokady,
- czas odpowiedzi podczas aktualizacji.

Domyślne ustawienie parametru monitora Timestamp to ON.

Zalecenie: Nadając parametrowi monitora Timestamp wartość OFF dla wszystkich aplikacji, można zwiększyć wydajność. Jeśli jednak dla jednej aplikacji parametr monitora Timestamp będzie miał wartość ON, system będzie kontynuował gromadzenie pomiarów czasów odpowiedzi. Dlatego wyłączenie parametru monitora Timestamp tylko dla niektórych aplikacji nie poprawi wydajności.

Wyłączenie tego parametru ma także inne konsekwencje.

- Wyłączenie parametru monitora Timestamp dla wszystkich aplikacji wymaga zatrzymania i restartowania instancji programu DB2 w celu wprowadzenia zmiany.
- Wyłączenie parametru monitora Timestamp powoduje wyłączenie zbierania informacji o datownikach dla aplikacji w systemie stowarzyszonym i dla innych aplikacji. Także lokalna baza danych nie będzie otrzymywała tych informacji.

Jeśli informacje o datownikach są potrzebne dla lokalnych aplikacji działających poza systemem stowarzyszonym, nie należy wyłączać parametru monitora Timestamp.

| Do nadania parametrowi Timestamp wartości OFF dla wszystkich aplikacji służy następująca
| komenda:

| `update dbm cfg using dft_mon_timestamp off`

| Następnie należy użyć komend:

| `db2stop`
| `db2start`

| Zatrzymanie i ponowne uruchomienie programu DB2 UDB spowoduje wyłączenie parametru
| dla wszystkich aplikacji.

| Szczegółowe informacje dotyczące wszystkich elementów monitorowanych za pomocą
| parametru Timestamp przedstawiono w osobnej sekcji.

| **Informacje pokrewne:**

- |
 - Rozdział 31, “Elementy monitora systemu stowarzyszonej bazy danych”, na stronie 307

Rozdział 12. Paralelizm i zapytania odwołujące się do pseudonimów

W tym rozdziale opisano sposób realizowania paralelizmu dla zapytań, które odwołują się do pseudonimów.

Ten rozdział zawiera:

- “Paralelizm i zapytania odwołujące się do pseudonimów”
- “Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów” na stronie 158
- “Włączanie paralelizmu wewnątrz partycji dla zapytań odwołujących się do pseudonimów” na stronie 158
- “Paralelizm między partycjami i zapytania odwołujące się do pseudonimów” na stronie 159
- “Włączanie paralelizmu między partycjami dla zapytań odwołujących się do pseudonimów” na stronie 161
- “Obliczeniowe grupy partycji” na stronie 162
- “Definiowanie obliczeniowej grupy partycji” na stronie 162
- “Paralelizm między partycjami dla zapytań odwołujących się do pseudonimów - oczekiwania w zakresie wydajności” na stronie 163
- “Paralelizm mieszany i zapytania odwołujące się do pseudonimów” na stronie 164
- “Włączanie paralelizmu mieszanego dla zapytań odwołujących się do pseudonimów” na stronie 164
- “Plany dostępu równoległego dla zapytań odwołujących się do pseudonimów” na stronie 165
- “Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 165
- “Paralelizm między partycjami i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 167
- “Paralelizm mieszany i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 169

Paralelizm i zapytania odwołujące się do pseudonimów

Zapytania, które zawierają pseudonimy, mogą uczestniczyć w trzech typach paralelizmu wewnątrz zapytań:

- paralelizmie zapytania wewnątrz partycji w konfiguracjach wieloprocesorowych z jedną partycją,
- paralelizmie zapytania wewnątrz partycji w konfiguracjach z wieloma partycjami,
- mieszanym paralelizmie zapytania, który składa się z paralelizmu wewnątrz partycji i między partycjami, w konfiguracjach z każdą partycją działającą na komputerze SMP.

Pojęcia pokrewne:

- “Parallelism” w podręczniku *Administration Guide: Planning*
- “Partition and processor environments” w podręczniku *Administration Guide: Planning*
- “The DB2 Process Model” w podręczniku *Administration Guide: Performance*
- “Obliczeniowe grupy partycji” na stronie 162

- “Plany dostępu równoległego dla zapytań odwołujących się do pseudonimów” na stronie 165
- “Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów” na stronie 158
- “Paralelizm między partycjami i zapytania odwołujące się do pseudonimów” na stronie 159
- “Paralelizm mieszany i zapytania odwołujące się do pseudonimów” na stronie 164

Informacje pokrewne:

- Rozdział 20, “Opcje opakowań w systemie stowarzyszonym”, na stronie 223

Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów

Paralelizm wewnątrz partycji to proces dzielenia zapytania na kilka współbieżnych części, które są uruchamiane równoległe przez wiele procesów na pojedynczej partycji bazy danych. W zapytaniach stowarzyszonych część zapytania, która dotyczy danych lokalnych, jest uruchamiana równoległe, a część dotycząca pseudonimu - szeregowo.

Zapytania odwołujące się do tabel lokalnych i pseudonimów mogą działać szybciej niż w poprzednich wersjach programu DB2[®] Information Integrator, ponieważ lokalne części zapytania mogą być wykonywane przez wiele procesorów.

Stopień paralelizmu wewnątrz partycji jest sterowany przez parametr konfiguracyjny DFT_DEGREE i rejestr specjalny CURRENT DEGREE.

Pojęcia pokrewne:

- “Parallelism” w podręczniku *Administration Guide: Planning*
- “Partition and processor environments” w podręczniku *Administration Guide: Planning*

Zadania pokrewne:

- “Enabling intra-partition parallelism for queries” w podręczniku *Administration Guide: Implementation*
- “Włączanie paralelizmu wewnątrz partycji dla zapytań odwołujących się do pseudonimów” na stronie 158

Informacje pokrewne:

- “Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 165

Włączanie paralelizmu wewnątrz partycji dla zapytań odwołujących się do pseudonimów

Dla zapytań, które odwołują się do lokalnych tabel i pseudonimów w środowisku wieloprocessorowym, można włączyć paralelizm wewnątrz partycji. Serwer stowarzyszony będzie wówczas mógł przetwarzać lokalne tabele równoległe.

Ograniczenia:

Serwer stowarzyszony może przetwarzać równoległe tylko lokalną część zapytania. Partycja koordynująca przetwarza wszystkie operacje na zdalnej części zapytania szeregowo.

Procedura:

1. Parametrowi konfiguracyjnemu bazy danych INTRA_PARALLEL nadaj wartość YES.
2. Parametrowi konfiguracyjnemu bazy danych MAX_QUERYDEGREE nadaj wartość większą niż 1.
3. Parametrowi konfiguracyjnemu bazy danych DFT_DEGREE nadaj wartość większą niż 1 lub ustaw rejestr specjalny CURRENT DEGREE.

Nadanie parametrowi DFT_DEGREE wartości ANY spowoduje, że domyślny stopień paralelizmu wewnątrz partycji będzie równy liczbie procesorów w komputerze.

Pojęcia pokrewne:

- “Paralelizm i zapytania odwołujące się do pseudonimów” na stronie 157

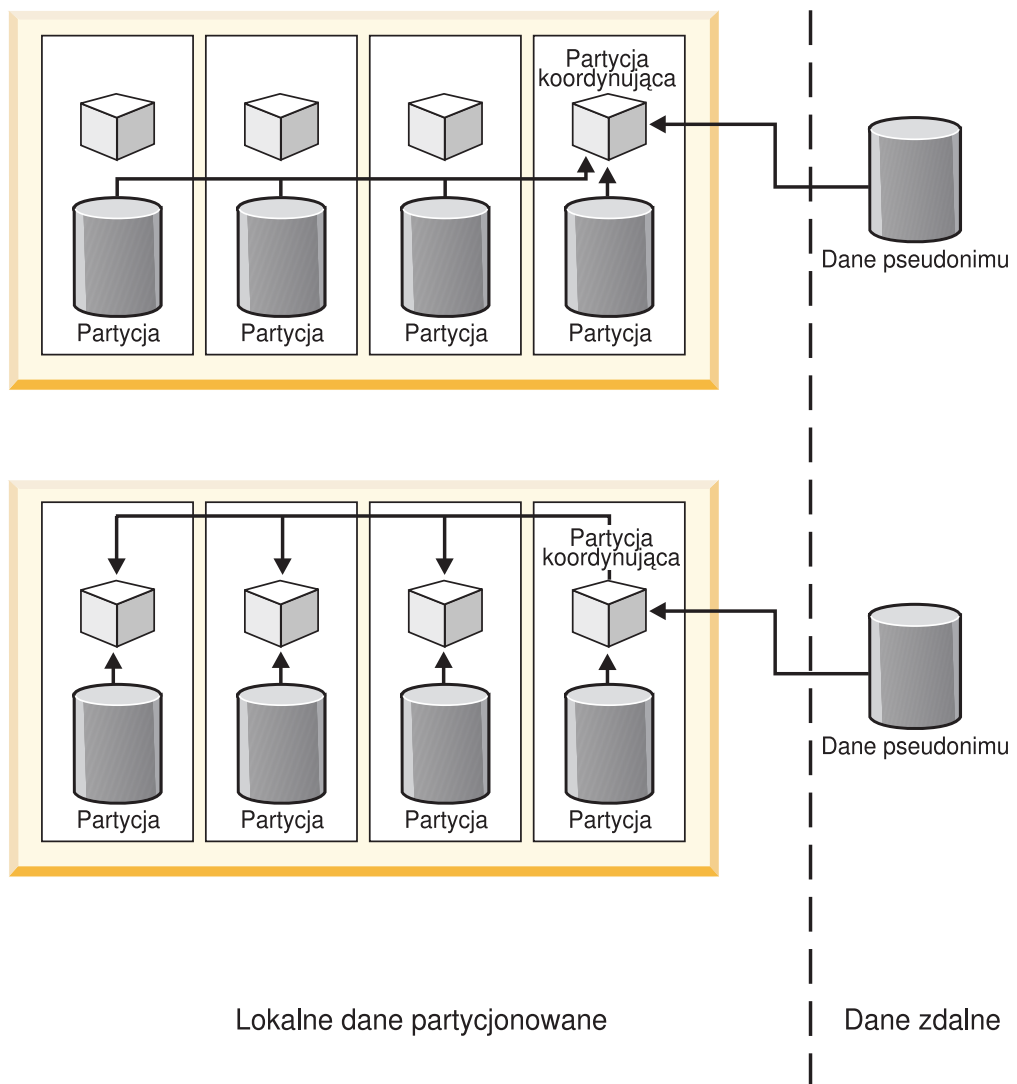
Informacje pokrewne:

- “ALTER WRAPPER statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE WRAPPER statement” w podręczniku *SQL Reference, Volume 2*

Paralelizm między partycjami i zapytania odwołujące się do pseudonimów

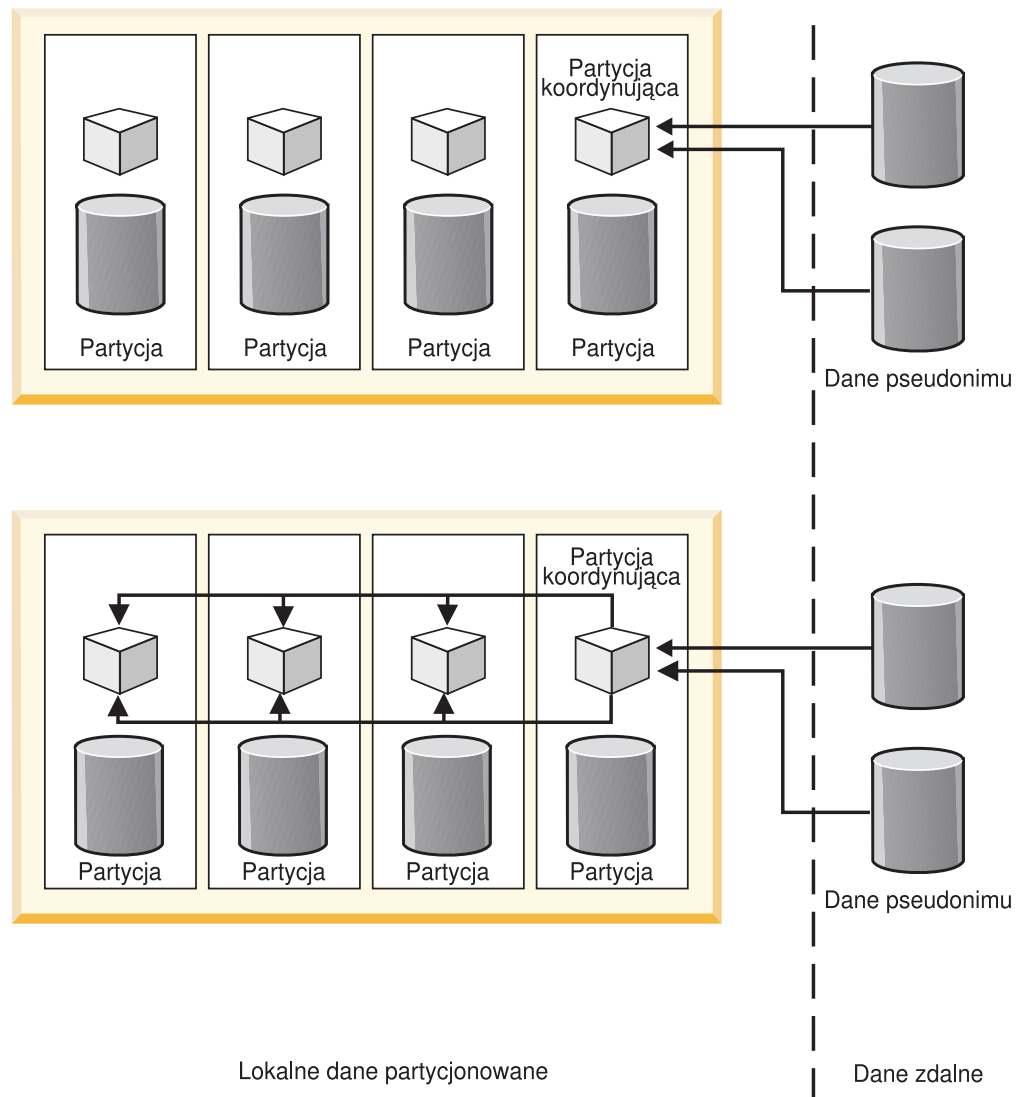
Paralelizm między partycjami to proces dzielenia pojedynczego zapytania na kilka części, które są uruchamiane równolegle na różnych partycjach partycjonowanej bazy danych.

W zapytaniach odwołujących się do danych lokalnych i zdalnych serwer stowarzyszony może umieścić zdalne dane na wszystkich partycjach lokalnych. Rys. 7 na stronie 160 przedstawia koncepcję paralelizmu między partycjami obejmującego lokalne i zdalne źródła danych. W górnej części rysunku przedstawiono sposób przetwarzania tego zapytania w poprzednich wersjach programu DB2[®] Information Integrator. Zdalne dane pseudonimu i lokalne dane partycjonowane były przetwarzane szeregowo w jednej partycji koordynującej. W dolnej części rysunku przedstawiono sposób przetwarzania takich zapytań w obecnej wersji programu DB2 Information Integrator. Baza danych rozprowadza dane pseudonimu na partycjach systemu lokalnego w celu przetwarzania równoległego.



Rysunek 7. Paralelizm między partycjami lokalnymi i zdalnymi źródeł danych

Rys. 8 na stronie 161 przedstawia koncepcję paralelizmu między partycjami obejmującego tylko zdalne źródła danych. W górnej części rysunku przedstawiono szeregowe przetwarzanie zdalnych danych pseudonimu w pojedynczej partycji koordynującej. W dolnej części rysunku przedstawiono rozprowadzanie danych przez partycję koordynującą na partycjach obliczeniowej grupy partycji.



Rysunek 8. Paralelizm między partycjami zapytań odwołujących się tylko do zdalnych źródeł danych

Pojęcia pokrewne:

- “Paralelizm i zapytania odwołujące się do pseudonimów” na stronie 157
- “Obliczeniowe grupy partycji” na stronie 162

Zadania pokrewne:

- “Włączanie paralelizmu między partycjami dla zapytań odwołujących się do pseudonimów” na stronie 161

Włączanie paralelizmu między partycjami dla zapytań odwołujących się do pseudonimów

Odwołujące się do pseudonimów zapytania, które można równoległe uruchamiać na wielu partycjach, mogą być wykonywane szybciej.

W środowisku partycjonowanej bazy danych zapytania stowarzyszone mogą być uruchamiane równoległe, gdy spełniają następujące warunki:

- obejmują kombinację pseudonimów i lokalnych tabel partycjonowanych,
- obejmują wyłącznie pseudonimy korzystające z obliczeniowej grupy partycji.

Aby korzystać z paralelizmu między partycjami, nie ma potrzeby ustawiania żadnych parametrów ani parametrów konfiguracyjnych bazy danych w środowisku partycjonowanym.

Ograniczenia:

Równolegle mogą być uruchamiane tylko te części zapytań odwołujących się do pseudonimów, które korzystają z opakowań chronionych. Części zapytań używające opakowań zaufanych nie mogą być uruchamiane równolegle.

Procedura:

1. Skonfiguruj obliczeniową grupę partycji.
2. Użyj instrukcji CREATE WRAPPER lub ALTER WRAPPER z opcją DB2_FENCED o wartości Y.

Pojęcia pokrewne:

- “Obliczeniowe grupy partycji” na stronie 162
- “Paralelizm między partycjami i zapytania odwołujące się do pseudonimów” na stronie 159

Obliczeniowe grupy partycji

Obliczeniowa grupa partycji definiuje zestaw partycji, którego ma używać optymalizator do dynamicznej operacji redystrybucji dla operacji łączenia. Obliczeniowa grupa partycji to grupa partycji bazy danych, inna niż IBMCATNODEGROUP, określona w katalogu systemowym SYSCAT.DBPARTITIONGROUPS.

Z obliczeniowych grup partycji można korzystać na potrzeby stowarzyszonego paralelizmu zapytań między partycjami. Dzięki nim można równolegle uruchamiać części pseudonimów w zapytaniach. W planie zapytania obejmującym obliczeniową grupę partycji serwer stowarzyszony dokonuje redystrybucji danych pseudonimu na partycje w celu utworzenia łączenia równoległego. Ten typ planu może zapewnić szybsze działanie zapytania, gdy objętość danych pseudonimu uczestniczących w łączeniu jest duża.

Do określenia obliczeniowej grupy partycji można użyć zmiennej rejestru DB2_COMPPARTITIONGROUP.

Pojęcia pokrewne:

- “Paralelizm i zapytania odwołujące się do pseudonimów” na stronie 157

Zadania pokrewne:

- “Definiowanie obliczeniowej grupy partycji” na stronie 162

Definiowanie obliczeniowej grupy partycji

Zdefiniowanie obliczeniowej grupy partycji umożliwi optymalizatorowi zastosowanie planu, w którym dane pseudonimu są redystrybuowane na partycje obliczeniowej grupy partycji. Obliczeniowe grupy partycji definiuje się w celu włączenia paralelizmu między partycjami dla zapytań lub ich części, które odwołują się wyłącznie do pseudonimów.

Wymagania wstępne:

Wszystkie grupy partycji reprezentujące obliczeniową grupę partycji dla wszystkich baz danych w instancji muszą mieć taką samą nazwę. W każdej bazie danych mogą one być zdefiniowane inaczej, ale ich nazwa musi być taka sama.

Ograniczenia:

Optymalizator użyje obliczeniowych grup partycji wyłącznie dla części zapytania, które odwołują się do pseudonimów, a nie do danych lokalnych.

Procedura:

Aby zdefiniować obliczeniową grupę partycji, uruchom następującą komendę w wierszu komend programu DB2:

```
db2set DB2_COMPPARTITIONGROUP=nazwa_grupy_partycji
```

`nazwa_grupy_partycji` to nazwa grupy partycji, która ma być zdefiniowana jako obliczeniowa grupa partycji.

Poniższy przykład przedstawia sposób definiowania obliczeniowej grupy partycji FINANCE3 przy użyciu zmiennej rejestru DB2_COMPPARTITIONGROUP.

```
db2set DB2_COMPPARTITIONGROUP=FINANCE3
```

Pojęcia pokrewne:

- “Obliczeniowe grupy partycji” na stronie 162

Informacje pokrewne:

- “db2set - DB2 Profile Registry Command” w podręczniku *Command Reference*

Paralelizm między partycjami dla zapytań odwołujących się do pseudonimów - oczekiwania w zakresie wydajności

W przypadku zapytań odwołujących się do kombinacji lokalnych tabel partycjonowanych i pseudonimów optymalizator może wybrać plan wykonania, w którym dane pseudonimu są redystrybuowane na odpowiednich partycjach. Plany z redystrybucją mogą przyspieszyć wykonywanie zapytań, gdy ilość danych pseudonimu włączeniu jest mniejsza niż ilość lokalnych danych partycjonowanych. Gdy ilość danych pseudonimu włączeniu jest znacząco większa niż ilość danych lokalnych, użycie planu z redystrybucją danych pseudonimu jest mało prawdopodobne. Jeśli optymalizator nie wybierze planu przetwarzania równoległego, serwer stowarzyszony wykona łączenia między pseudonimami i tabelami lokalnymi szeregowo w partycji koordynującej.

W przypadku łączeń między dwoma pseudonimami plan wykonania, w którym dane są rozprowadzane między wszystkimi partycjami obliczeniowej grupy partycji, może być korzystny, o ile obejmuje dużą ilość danych. Zalety równoległego przetwarzania dużych łączeń przeważają nad dodatkowymi kosztami redystrybucji danych między wieloma partycjami. Jeśli ilość danych pseudonimu jest względnie mała, łączenie nie jest na tyle kosztowne, aby dodatkowy koszt redystrybucji danych między partycjami uznać za uzasadniony. W ogólności optymalizator wybiera plany z obliczeniową grupą partycji, gdy przetwarzane pseudonimy są duże; w przeciwnym razie serwer stowarzyszony łączy pseudonimy szeregowo w partycji koordynującej.

Pojęcia pokrewne:

- “Obliczeniowe grupy partycji” na stronie 162
- “Paralelizm między partycjami i zapytania odwołujące się do pseudonimów” na stronie 159

Informacje pokrewne:

- “Paralelizm między partycjami i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 167

Paralelizm mieszany i zapytania odwołujące się do pseudonimów

Dla zapytań, które odwołują się do lokalnych tabel i pseudonimów w środowisku partycjonowanym, można korzystać jednocześnie z paralelizmu wewnątrz partycji i między partycjami. Serwer stowarzyszony może rozprowadzić dane między partycjami i przetwarzać je równoległe w ramach każdej partycji.

Pojęcia pokrewne:

- “Paralelizm i zapytania odwołujące się do pseudonimów” na stronie 157

Zadania pokrewne:

- “Włączanie paralelizmu mieszanego dla zapytań odwołujących się do pseudonimów” na stronie 164

Włączanie paralelizmu mieszanego dla zapytań odwołujących się do pseudonimów

Zapytania stowarzyszone uruchamiane równoległe działają szybciej. Używając paralelizmu wewnątrz partycji i między partycjami można poprawić wydajność zapytań, które odwołują się do danych lokalnych i zdalnych. Stowarzyszone zapytania odwołują się do kombinacji pseudonimów i lokalnych tabel partycjonowanych.

Procedura:

1. Parametrowi konfiguracyjnemu bazy danych MAX_QUERYDEGREE nadaj wartość większą niż 1.
2. Parametrowi konfiguracyjnemu bazy danych DFT_DEGREE nadaj wartość większą niż 1 lub ustaw rejestr specjalny CURRENT DEGREE. Nadanie parametrowi DFT_DEGREE wartości ANY spowoduje, że domyślny stopień paralelizmu wewnątrz partycji będzie równy liczbie procesorów SMP w komputerze.
3. Skonfiguruj obliczeniową grupę partycji.
4. Użyj instrukcji CREATE WRAPPER lub ALTER WRAPPER z opcją DB2_FENCED o wartości Y.

Pojęcia pokrewne:

- “Obliczeniowe grupy partycji” na stronie 162
- “Paralelizm mieszany i zapytania odwołujące się do pseudonimów” na stronie 164

Zadania pokrewne:

- “Trusted and fenced mode process environments” w podręczniku *IBM DB2 Information Integrator Wrapper Developer's Guide*

Informacje pokrewne:

- “Paralelizm mieszany i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 169

Plany dostępu równoległego dla zapytań odwołujących się do pseudonimów

Informacje o planach dostępu używanych przez optymalizator podczas przetwarzania zapytań są przechwytywane przez narzędzie SQL Explain.

Pojęcia pokrewne:

- “SQL explain facility” w podręczniku *Administration Guide: Performance*
- “Explain tools” w podręczniku *Administration Guide: Performance*
- “SQL explain tools” w podręczniku *Administration Guide: Performance*

Informacje pokrewne:

- “Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 165
- “Paralelizm między partycjami i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 167
- “Paralelizm mieszany i zapytania odwołujące się do pseudonimów - przykłady planów dostępu” na stronie 169

Paralelizm wewnątrz partycji i zapytania odwołujące się do pseudonimów - przykłady planów dostępu

Narzędzie DB2 UDB Explain może wygenerować plan dostępu, którego użyje optymalizator podczas przetwarzania zapytania. W poniższym przykładzie pokazano, w jaki sposób optymalizator uzyskuje dostęp do danych pseudonimu w środowisku paralelizmu wewnątrz partycji.

Przykład 1: Obsługa paralelizmu w wersji programu DB2 Information Integrator wcześniejszej niż 8.2

W tym przykładzie serwer stowarzyszony szeregowo przetwarza łączenie tabeli lokalnej ORDERS i pseudonimu ITEMS. Paralelizm wewnątrz partycji nie jest używany.

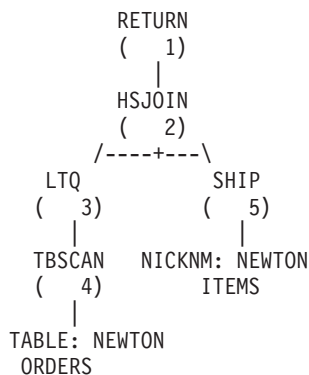
```
SELECT *  
FROM ORDERS A, ITEMS B  
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```

```
          RETURN  
          ( 1)  
          |  
        HSJOIN  
          ( 2)  
    /-----\  
   TBSCAN      SHIP  
   ( 3)        ( 4)  
   |           |  
TABLE: NEWTON NICKNM: NEWTON  
ORDERS        ITEMS
```

Przykład 2: Obsługa paralelizmu w wersji 8.2 programu DB2 Information Integrator

W tym przykładzie łączenie zapytanie może zostać wykonane szybciej, dzięki równoległemu odczytowi tabeli lokalnej przed szeregowym łączeniem z pseudonimem.

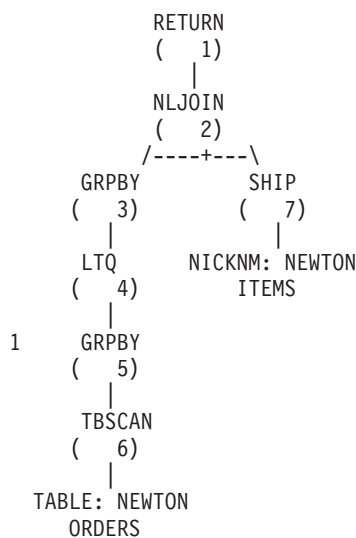
```
SELECT *
FROM ORDERS A, ITEMS B
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



Przykład 3: Paralelizm wewnątrz partycji z agregacją

W tym przykładzie baza danych agreguje równolegle lokalną tabelę danych w partycji, poprawiając wydajność agregacji. Łączenie tabeli lokalnej i pseudonimu odbywa się szeregowo w węźle koordynującym.

```
SELECT *
FROM ITEMS A
WHERE ID =
  (SELECT MAX(ID)
   FROM ORDERS
   WHERE NUMBER = 10)
```



Pojęcia pokrewne:

- “Example two: single-partition plan with intra-partition parallelism” w podręczniku *Administration Guide: Performance*
- “Plany dostępu równoległego dla zapytań odwołujących się do pseudonimów” na stronie 165

Zadania pokrewne:

- “Włączanie paralelizmu wewnątrz partycji dla zapytań odwołujących się do pseudonimów” na stronie 158

Paralelizm między partycjami i zapytania odwołujące się do pseudonimów - przykłady planów dostępu

Narzędzie DB2 UDB Explain może wygenerować plan dostępu, którego użyje optymalizator podczas przetwarzania zapytania. W poniższym przykładzie pokazano, w jaki sposób optymalizator uzyskuje dostęp do danych pseudonimu w środowisku paralelizmu między partycjami.

Przykład 1: Tryb zaufany

W tym przykładzie pseudonim używa zaufanego opakowania. Baza danych wykonuje szeregowo operację łączenia tabeli lokalnej z pseudonimem w partycji koordynującej. Baza danych scala na partycji koordynującej dane lokalne rozproszone na dwóch partycjach. Następnie serwer stowarzyszony łączy dane lokalne z danymi pseudonimu. Baza danych szeregowo łączy w partycji koordynującej pseudonimy zdefiniowane przy użyciu opakowania zaufanego. Baza danych nie może rozprowadzić danych na wielu partycjach w celu utworzenia łączenia równoległego.

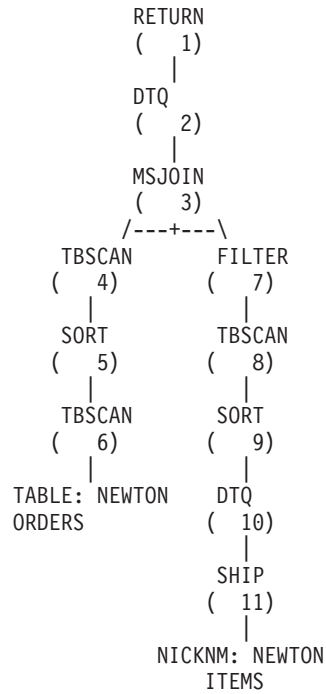
```
SELECT *
FROM ORDERS A, ITEMS B
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```

```
          RETURN
          ( 1)
          |
          HSJOIN
          ( 2)
          /-----\
         DTQ       SHIP
        ( 3)       ( 5)
         |         |
         TBSCAN   NICKNM: NEWTON
        ( 4)       ITEMS
         |
        TABLE: NEWTON
        ORDERS
```

Przykład 2: Tryb chroniony

W tym przykładzie pseudonim używa opakowania chronionego. Serwer stowarzyszony rozprowadza dane pseudonimu na inne partycje i wykonuje łączenie z danymi lokalnymi równoległe.

```
SELECT *
FROM ORDERS A, ITEMS B
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```

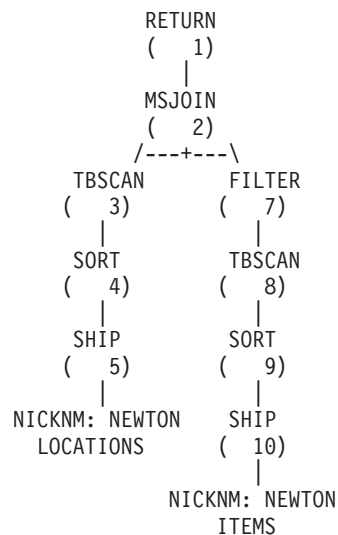


Przykład 3: Tryb chroniony bez obliczeniowej grupy partycji

W tym przykładzie dwa pseudonimy używają opakowania chronionego i nie ma zdefiniowanej obliczeniowej grupy partycji. Serwer stowarzyszony wykonuje łączenie w partycji koordynującej. Serwer stowarzyszony nie rozprawdza danych na inne partycje w celu przetwarzania.

```

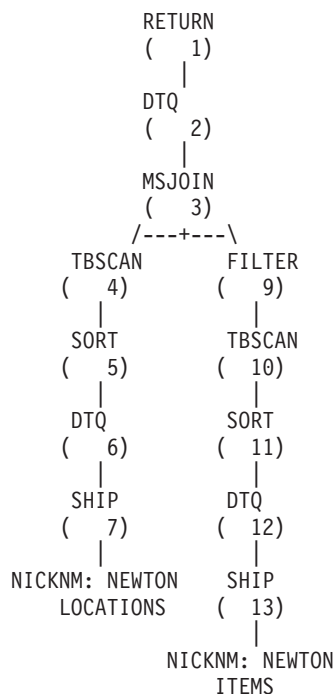
SELECT *
FROM ITEMS A, LOCATIONS B
WHERE A.ID1 = B.ID1
  
```



Przykład 4: Tryb chroniony z obliczeniową grupą partycji

W tym przykładzie pseudonimy używają opakowań chronionych i jest zdefiniowana obliczeniowa grupa partycji. W tej sytuacji optymalizator wybiera plan, zgodnie z którym dane są rozprowadzane z partycji koordynującej na inne partycje obliczeniowej grupy partycji.

```
SELECT *  
FROM ITEMS A, LOCATIONS B  
WHERE A.ID = B.ID
```



Pojęcia pokrewne:

- “Example three: multipartition plan with inter-partition parallelism” w podręczniku *Administration Guide: Performance*
- “Plany dostępu równoległego dla zapytań odwołujących się do pseudonimów” na stronie 165

Zadania pokrewne:

- “Trusted and fenced mode process environments” w podręczniku *IBM DB2 Information Integrator Wrapper Developer’s Guide*
- “Włączanie paralelizmu między partycjami dla zapytań odwołujących się do pseudonimów” na stronie 161

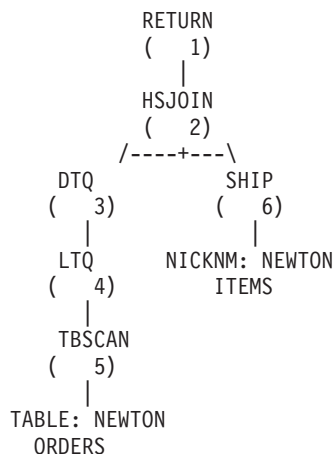
Paralelizm mieszany i zapytania odwołujące się do pseudonimów - przykłady planów dostępu

Narzędzie DB2 UDB Explain może wygenerować plan dostępu, którego użyje optymalizator podczas przetwarzania zapytania. W poniższym przykładzie pokazano, w jaki sposób optymalizator uzyskuje dostęp do danych pseudonimu w środowisku, w którym wykorzystuje się zarówno paralelizm wewnątrz partycji, jak i między partycjami.

Przykład 1: Tryb zaufany

Poniższy przykład przedstawia łączenie między tabelą lokalną i pseudonimem w trybie zaufanym. Serwer stowarzyszony przetwarza dane lokalne równolegle w każdej partycji, a następnie wykonuje ich łączenie z danymi pseudonimu w partycji koordynującej. Serwer stowarzyszony nie przetwarza danych pseudonimu równolegle na różnych partycjach albo przy użyciu różnych procesorów danej partycji.

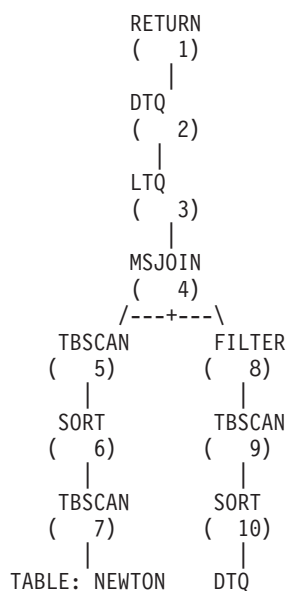
```
SELECT *  
FROM ORDERS A, ITEMS B  
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



Przykład 2: Tryb chroniony

Poniższy przykład przedstawia łączenie między tabelą lokalną i pseudonimem w trybie chronionym. Serwer stowarzyszony rozprawdza dane pseudonimu z partycji koordynującej na inne partycje w systemie. Serwer stowarzyszony przetwarza dane tabeli lokalnej równolegle, zarówno przy użyciu różnych partycji, jak i procesorów.

```
SELECT *  
FROM ORDERS A, ITEMS B  
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



```
ORDERS      ( 11)
             |
             SHIP
             ( 12)
             |
NICKNM: NEWTON
ITEMS
```

Pojęcia pokrewne:

- “Example four: multipartition plan with inter-partition and intra-partition parallelism” w podręczniku *Administration Guide: Performance*
- “Plany dostępu równoległego dla zapytań odwołujących się do pseudonimów” na stronie 165

Zadania pokrewne:

- “Trusted and fenced mode process environments” w podręczniku *IBM DB2 Information Integrator Wrapper Developer’s Guide*
- “Włączanie paralelizmu mieszanego dla zapytań odwołujących się do pseudonimów” na stronie 164

Rozdział 13. Zmaterializowane tabele zapytań i systemy stowarzyszone

W tym rozdziale opisano sposób korzystania ze zmaterializowanych tabel zapytań, które odwołują się do pseudonimów, w systemie stowarzyszonym.

Ten rozdział zawiera:

- “Zmaterializowane tabele zapytań i systemy stowarzyszone – przegląd”
- “Tworzenie stowarzyszonej zmaterializowanej tabeli zapytania” na stronie 174
- “Ograniczenia poszczególnych źródeł danych dotyczące zmaterializowanych tabel zapytań” na stronie 175
- “Ograniczenia dotyczące korzystania ze zmaterializowanych tabel zapytań z pseudonimami” na stronie 177

Zmaterializowane tabele zapytań i systemy stowarzyszone – przegląd

Zmaterializowana tabela zapytania to tabela, w której przechowywane są wyniki zapytania. Podczas ponownego uruchomienia zapytania mechanizm bazy danych może zwrócić dane ze zmaterializowanej tabeli zapytania. Z tabel tych - oraz z pseudonimów - można korzystać do poprawienia wydajności zapytania oraz w celu hermetyzowania części logiki. Ze zmaterializowanych tabel zapytań można także korzystać podczas tworzenia tabel buforowanych.

Efektywność zapytania uruchomionego w odniesieniu do zmaterializowanej tabeli zapytania w porównaniu z zapytaniem skierowanym do tabel lub pseudonimów bazowych określa optymalizator SQL. Optymalizator wybiera zmaterializowaną tabelę zapytania na podstawie następujących czynników:

- Zmaterializowana tabela zapytania musi być zgodna z całym zapytaniem lub jego częścią.
- Musi być spełnione kryterium odświeżania.
- Plan dostępu z użyciem zmaterializowanej tabeli zapytania musi być tańszy niż plan dostępu z użyciem tabel lub pseudonimów bazowych.

Zmaterializowane tabele zapytań są obsługiwane przez następujące źródła danych:

- Relacyjne źródła danych
 - DRDA[®]
 - Informix[®]
 - ODBC
 - Oracle
 - Sybase
 - MS SQL Server
 - Teradata
- Nierelacyjne źródła danych
 - BioRS
 - BLAST
 - Documentum
 - Entrez
 - Excel
 - HMMER
 - IBM[®] Lotus[®] Extended Search
 - Pliki o strukturze tabeli

- Usługi WWW
- WebSphere® Business Integration
- XML

Pojęcia pokrewne:

- “Tabele buforowane” na stronie 179
- “Performance and tuning planning— materialized query tables in a federated system” w podręczniku *IBM DB2 Information Integrator Application Developer’s Guide*

Zadania pokrewne:

- “Creating a materialized query table” w podręczniku *Administration Guide: Implementation*
- “Altering materialized query table properties” w podręczniku *Administration Guide: Implementation*
- “Tworzenie stowarzyszonej zmaterializowanej tabeli zapytania” na stronie 174

Informacje pokrewne:

- “Ograniczenia poszczególnych źródeł danych dotyczące zmaterializowanych tabel zapytań” na stronie 175

Tworzenie stowarzyszonej zmaterializowanej tabeli zapytania

Zmaterializowane tabele zapytań służą do lokalnego buforowania danych oraz do poprawienia wydajności zapytań. Do tworzenia tych tabel można używać pseudonimów obiektów z relacyjnych i nierelacyjnych źródeł danych.

Ograniczenia:

- Ograniczenia poszczególnych źródeł danych dotyczące zmaterializowanych tabel zapytań.
- Jeśli zapytanie ma w predykacie lub na liście wyboru szablon funkcji, musi on być częścią zmaterializowanej tabeli zapytania.

Procedura:

Aby utworzyć zmaterializowaną tabelę zapytania, należy użyć instrukcji CREATE TABLE dla pseudonimu reprezentującego obiekt ze zdalnego źródła danych.

Pojęcia pokrewne:

- “Zmaterializowane tabele zapytań i systemy stowarzyszone – przegląd” na stronie 173

Zadania pokrewne:

- “Creating a materialized query table” w podręczniku *Administration Guide: Implementation*
- “Dodawanie zmaterializowanej tabeli zapytania do tabeli buforowanej” na stronie 183

Informacje pokrewne:

- “Ograniczenia poszczególnych źródeł danych dotyczące zmaterializowanych tabel zapytań” na stronie 175
- “ALTER TABLE statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE TABLE statement” w podręczniku *SQL Reference, Volume 2*

Ograniczenia poszczególnych źródeł danych dotyczące zmaterializowanych tabel zapytań

W tym temacie opisano ograniczenia dotyczące tworzenia zmaterializowanych tabel zapytań dla następujących źródeł danych:

- Bio-RS
- BLAST
- Entrez
- HMMER
- IBM Lotus Extended Search
- Pliki o strukturze tabeli
- Usługi WWW
- XML

Ograniczenia źródeł danych Bio-RS, Entrez i IBM Lotus Extended Search

Te opakowania wymagają co najmniej jednego predykatu w klauzuli WHERE. Konieczne jest utworzenie zmaterializowanej tabeli zapytania, która spełnia wymagania opakowań w zakresie predykatów. Jeśli nie określi się predykatu, odświeżenie zmaterializowanej tabeli zapytania nie powiedzie się.

Ograniczenia źródeł danych BLAST i HMMER

Gdy korzysta się z opakowań BLAST i HMMER, źródła danych wymagają wartości dla niektórych predykatów. Opakowanie wprowadza pewne wartości domyślne. Dla innych predykatów należy określić wartość samodzielnie. Jeśli nie określi się predykatu, odświeżenie zmaterializowanej tabeli zapytania nie powiedzie się.

W poniższym przykładzie zmaterializowana tabela zapytania została utworzona pomyślnie, ponieważ opakowanie nie korzystało ze źródła danych.

```
CREATE TABLE MY_MQT AS (SELECT * FROM BLAST_NICK)
  DATA INITIALLY DEFERRED REFRESH DEFERRED ENABLE QUERY OPTIMIZATION;
```

Jednak podczas odświeżania tabeli miało miejsce połączenie ze źródłem danych BLAST w celu pobrania danych. Źródło danych BLAST zgłosiło błąd, ponieważ wymaga ono predykatu dla kolumny blast_seq.

W przypadku predykatów opcjonalnych, jeśli uruchomi się zapytanie z wartościami innymi niż domyślne, podczas tworzenia zmaterializowanej tabeli zapytania należy podać te wartości. Jeśli nie określi się predykatu, odświeżenie zmaterializowanej tabeli zapytania nie powiedzie się.

W poniższym przykładzie podczas tworzenia zmaterializowanej tabeli zapytania opakowanie nadaje parametrom opcjonalnym wartości domyślne. Te opcjonalne parametry to in_arg1 = 2 i in_arg2 < 30.

```
CREATE TABLE MY_MQT AS (SELECT * FROM BLAST_NICK WHERE BLAST_SEQ = '12345')
  DATA INITIALLY DEFERRED REFRESH DEFERRED ENABLE QUERY OPTIMIZATION;
```

Podczas uruchomienia poniższego zapytania optymalizator sprawdzi zgodność zapytania ze zmaterializowaną tabelą zapytania. Jeśli optymalizator wybierze plan dostępu z użyciem zmaterializowanej tabeli zapytania, wyniki tego zapytania będą niepoprawne. Będzie to spowodowane tym, że zmaterializowana tabela zapytania zawiera dane dla predykatu in_arg1 = 2, a zapytanie żądało danych z predykatem in_arg1 = 3.

```
SELECT * FROM BLAST_NICK WHERE BLAST_SEQ = '12345' AND IN_ARG1 = 3;
```

Po utworzeniu zmaterializowanej tabeli zapytania konieczne jest jawne określenie wartości pewnych predykatów, nawet jeśli opakowanie używa dla nich wartości domyślnych. W przeciwnym razie optymalizator może nie skierować zapytania do zgodnej zmaterializowanej tabeli zapytania. Może się to zdarzyć, ponieważ podczas tworzenia zmaterializowanej tabeli zapytania nie określono wartości tych predykatów w sposób jawny.

W poniższym przykładzie przedstawiono sytuację, w której optymalizator nie wybiera zmaterializowanej tabeli zapytania pomimo tego, że jest ona zgodna z zapytaniem. Jeśli zmaterializowana tabela zapytania zawiera jawnie ustaloną kolumnę wejściową z wartością domyślną a w zapytaniu predykat ten nie został określony, zapytanie nie zostanie skierowane do zmaterializowanej tabeli zapytania.

```
CREATE TABLE K55ADMIN.BLAST_NICK1_M1 AS (  
  SELECT SCORE, E_VALUE, QUERYSTRANDS  
  FROM K55ADMIN.BLAST_NICK1  
  WHERE BLASTSEQ='ATGATCGGATCGAATTCGAT'  
  AND E_VALUE < 10) DATA INITIALLY DEFERRED REFRESH DEFERRED;
```

Poniższe zapytanie po uruchomieniu nie zostanie skierowane do zmaterializowanej tabeli zapytania. Optymalizator wykryje, że w zmaterializowanej tabeli zapytania nie określono predykatu E_VALUE < 10.

```
SELECT SCORE, E_VALUE, QUERYSTRANDS  
FROM K55ADMIN.BLAST_NICK1 WHERE BLASTSEQ='ATGATCGGATCGAATTCGAT';
```

Ograniczenia plików o strukturze tabeli

Gdy dla pliku o strukturze tabeli zdefiniuje się pseudonim z opcją DOCUMENT, zmaterializowana tabela zapytania musi mieć predykat określający ścieżkę do pliku. Jeśli nie określi się predykatu, odświeżenie zmaterializowanej tabeli zapytania nie powiedzie się.

Ograniczenia usług WWW

Zmaterializowaną tabelę zapytania można utworzyć tylko w odniesieniu do spłaszczonego widoku hierarchii pseudonimów. Nie można utworzyć takiej tabeli dla każdego pseudonimu w hierarchii.

Ograniczenia XML

Nie można utworzyć zmaterializowanej tabeli zapytania dla tabeli podrzędnej.

Gdy dla tabeli XML zdefiniuje się pseudonim z opcją DOCUMENT, zmaterializowana tabela zapytania wymaga predykatu określającego ścieżkę do pliku. Jeśli nie określi się predykatu, odświeżenie zmaterializowanej tabeli zapytania nie powiedzie się.

Zadania pokrewne:

- “Tworzenie stowarzyszonej zmaterializowanej tabeli zapytania” na stronie 174

Informacje pokrewne:

- “Ograniczenia dotyczące korzystania ze zmaterializowanych tabel zapytań z pseudonimami” na stronie 177

Ograniczenia dotyczące korzystania ze zmaterializowanych tabel zapytań z pseudonimami

W środowisku partycjonowanej bazy danych program DB2 Information Integrator nie zezwala na używanie obsługiwanych przez system zmaterializowanych tabel zapytań, które odwołują się do pseudonimów.

Aby obejść to ograniczenie, można użyć zmaterializowanych tabel zapytań obsługiwanych przez użytkownika.

Na przykład dla nierelacyjnego pseudonimu o nazwie DEPART można użyć poniższych komend, aby zasymulować obsługiwaną przez system zmaterializowaną tabelę zapytania.

```
SET CURRENT MAINTAINED TABLE TYPES FOR OPTIMIZATION ALL;  
  
CREATE TABLE AST1(C1, C2)  
  AS (SELECT EMPNO, FIRSTNME FROM DEPART WHERE EMPNO>'000000')  
  DATA INITIALLY DEFERRED REFRESH DEFERRED  
  ENABLE QUERY OPTIMIZATION MAINTAINED BY USER;  
  
SET INTEGRITY FOR AST1 ALL IMMEDIATE UNCHECKED;  
  
INSERT INTO AST1 (SELECT EMPNO, FIRSTNME FROM DEPART WHERE EMPNO>'000000');  
  
SET CURRENT REFRESH AGE ANY;
```

Poniższa instrukcja wyboru może być skierowana do zdefiniowanej wcześniej zmaterializowanej tabeli zapytania:

```
SELECT EMPNO, FIRSTNME FROM DEPART  
  WHERE EMPNO > '000000' AND FIRSTNME LIKE 'AN%';
```

Rozdział 14. Tabele buforowane w systemie stowarzyszonym

W tym rozdziale opisano, co to jest tabela buforowana i jak z niej korzystać w systemie stowarzyszonym.

Ten rozdział zawiera:

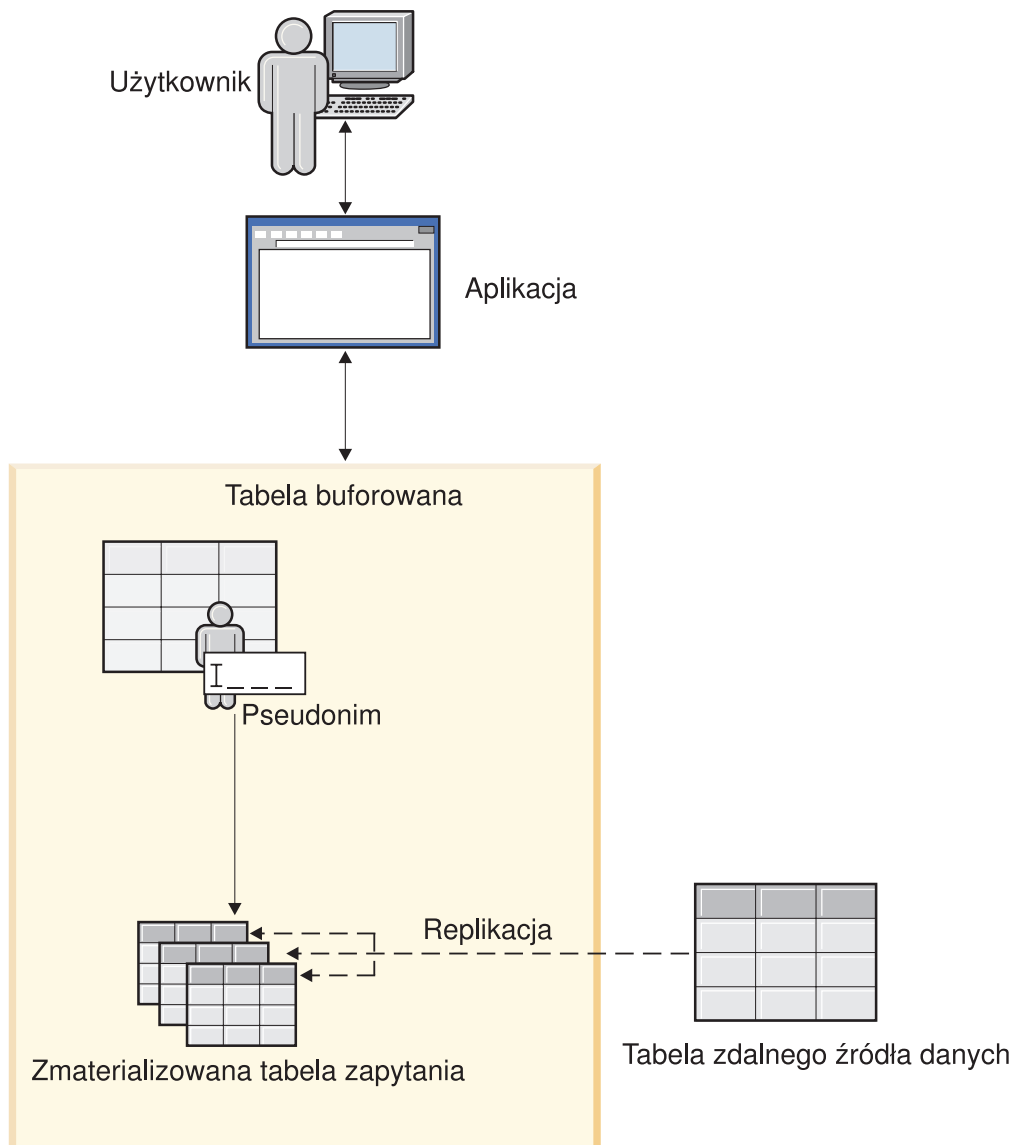
- “Tabele buforowane”
- “Tworzenie tabeli buforowanej” na stronie 181
- “Włączanie pamięci podręcznej” na stronie 182
- “Dodawanie zmaterializowanej tabeli zapytania do tabeli buforowanej” na stronie 183
- “Usuwanie zmaterializowanej tabeli zapytania z tabeli buforowanej” na stronie 183
- “Usuwanie tabeli buforowanej” na stronie 184

Tabele buforowane

Tabela buforowana może poprawić wydajność zapytania, dzięki udostępnieniu lokalnego podzbioru danych i wyeliminowaniu konieczności bezpośredniego dostępu do zdalnego relacyjnego źródła danych. Tabela buforowana składa się z następujących komponentów:

- Pseudonimu w systemie stowarzyszonej bazy danych z takimi samymi definicjami kolumn i prawami dostępu do danych, jak zdalna tabela w relacyjnym źródle danych.
- Co najmniej jednej obsługiwanej przez użytkownika zmaterializowanej tabeli zapytania zdefiniowanej dla pseudonimu. Pseudonim może zawierać podzbiór najczęściej używanych danych ze zdalnego źródła danych.
- Zdefiniowanego przez użytkownika harmonogramu replikacji skojarzonego z każdą zmaterializowaną tabelą zapytania w celu jej synchronizacji ze zdalnym źródłem danych.

Rys. 9 na stronie 180 ilustruje koncepcję tabeli buforowanej.



Rysunek 9. Tabele buforowane składają się z pseudonimu, zmaterializowanych tabel zapytań i harmonogramu replikacji

Tabela buforowana ma taką samą nazwę, jak wchodzący w jej skład pseudonim. Tabelę buforowaną można skojarzyć tylko z jedną tabelą zdaną. Tabela buforowana może być pełną repliką tabeli ze zdanego źródła danych lub podzbiorem wierszy z tej tabeli. Tabela buforowana zawiera dane lokalne definiowane przez skojarzone z nią zmaterializowane tabele zapytań. Podstawowa różnica między tabelą buforowaną a zmaterializowaną tabelą zapytania polega na tym, że tabela buforowana jest konfigurowana z poziomu Centrum sterownia DB2®.

Aplikacje kierujące zapytania do zdanego źródła danych mogą po wprowadzeniu niewielkich zmian kierować zapytania do tabeli buforowanej. Podczas przetwarzania zapytania optymalizator skieruje je do tabeli buforowanej lub do zdalnej tabeli w relacyjnym źródle danych.

Pojęcia pokrewne:

- “Replicated materialized query tables” w podręczniku *Administration Guide: Planning*
- “Zmaterializowane tabele zapytań i systemy stowarzyszone – przegląd” na stronie 173

Zadania pokrewne:

- “Tworzenie tabeli buforowanej” na stronie 181
- “Tworzenie stowarzyszonej zmaterializowanej tabeli zapytania” na stronie 174

Tworzenie tabeli buforowanej

Tabeli buforowanej można użyć do przechowywania często używanych danych, które nie zmieniają się zbyt często. Dzięki stosowaniu tabel buforowanych można poprawić wydajność zapytań.

Dane w tabeli buforowanej mogą pochodzić ze źródeł danych, dla których program DB2 UDB obsługuje replikację heterogeniczną. Takie źródła danych to produkty z rodziny DB2 oraz systemy baz danych Informix, Microsoft SQL Server, Oracle i Sybase.

Wymagania wstępne:

- Nadaj parametrowi konfiguracyjnemu menedżera bazy danych dla serwera DB2 UDB, FEDERATED, wartość YES.
- Aby korzystać ze źródeł danych Informix, zainstaluj i skonfiguruj oprogramowanie Informix Client SDK na serwerze stowarzyszonym.
- Aby umieszczać w tabelach pamięci podręcznej dane z tabel programu DB2 UDB dla systemów Linux, UNIX i Windows, w bazie danych, w której przechowywane są te tabele, skonfiguruj *protokołowanie z przechowywaniem protokołów*. Protokołowanie z przechowywaniem protokołów to rodzaj protokołowania archiwalnego. Aby to zrobić, nadaj parametrowi LOGRETAIN wartość RECOVER.
- Stowarzyszona baza danych lub źródłowa baza danych musi znajdować się na komputerze, na którym tworzysz tabele buforowane. Jeśli stowarzyszona baza danych ani źródłowa baza danych nie są lokalne, wpisz je do katalogu na komputerze lokalnym. Nazwa aliasu użyta podczas wpisywania bazy danych do katalogu musi być taka sama, jak nazwa bazy danych.
- Identyfikator użytkownika w odwzorowaniu użytkownika między bazami danych musi mieć uprawnienie do tworzenia tabel w źródłowej bazie danych.

Procedura:

Aby utworzyć tabelę buforowaną:

1. W Centrum sterowania DB2 rozwiń folder **Buforuj obiekty**.
2. Kliknij prawym przyciskiem myszy folder **Buforuj tabele** i kliknij polecenie **Utwórz**. Zostanie otwarty Kreator tabeli buforowanej.
3. Wykonaj instrukcje kreatora.

Podczas tworzenia tabeli buforowanej Kreator tabeli buforowanej tworzy jedną zmaterializowaną tabelę zapytania. Aby przechowywać inne dane z tego samego źródła danych, można utworzyć dodatkowe zmaterializowane tabele zapytań.

Pojęcia pokrewne:

- “Tabele buforowane” na stronie 179

Zadania pokrewne:

- “Włączanie pamięci podręcznej” na stronie 182
- “Dodawanie zmaterializowanej tabeli zapytania do tabeli buforowanej” na stronie 183

Informacje pokrewne:

- “logretain - Log retain enable configuration parameter” w podręczniku *Administration Guide: Performance*

Włączanie pamięci podręcznej

Włączenie pamięci podręcznej powoduje aktywowanie programów przechwytyjących i wprowadzających zmiany. Te programy replikują dane ze źródła danych do zmaterializowanej tabeli zapytania.

W ogólności, utworzenie tabeli buforowanej powoduje włączenie pamięci podręcznej. Jednak w wymienionych poniżej sytuacjach konieczne może być ręczne włączenie pamięci podręcznej:

- Gdy tworzy się tabelę buforowaną i spełnione są obydwa poniższe warunki, baza danych nie włącza pamięci podręcznej na potrzeby replikacji:
 - źródłem danych jest tabela programu DB2 UDB dla systemu Linux, UNIX lub Windows;
 - w bazie danych, w której rezyduje tabela, nie skonfigurowano protokołowania z przechowywaniem protokołów.
- Po utworzeniu tabeli buforowanej pamięć podręczna jest wyłączana, gdy program przechwytyjący lub wprowadzający zmiany zostanie zatrzymany. Po restartowaniu programu przechwytyjącego lub wprowadzającego zmiany w Centrum replikacji DB2 konieczne jest włączenie pamięci podręcznej, aby można było rozpocząć replikację.

W Centrum sterowania DB2 *protokołowanie z przechowywaniem protokołów* jest niekiedy określane terminem *protokołowanie archiwalne*.

Wymagania wstępne:

Jeśli pamięć podręczna jest włączana dla tabeli z bazy danych programu DB2 UDB dla systemów Linux, UNIX lub Windows, muszą być spełnione następujące wymagania wstępne:

- W bazie danych, w której znajduje się tabela, musi być skonfigurowane protokołowanie z przechowywaniem protokołów.
- Musi być uruchomiony serwer DAS (Database Administration Server).

Procedura:

Aby włączyć pamięć podręczną:

1. W Centrum sterowania DB2 rozwiń w drzewie obiektów folder **Buforuj obiekty**, a następnie folder **Tabele buforowane**.
2. Kliknij prawym przyciskiem myszy tabelę pamięci podręcznej z pamięcią podręczną, którą chcesz włączyć, i kliknij polecenie **Włącz buforowanie**. Zostanie otwarte okno **Włącz buforowanie**.
3. Zaznacz pamięć podręczną, którą chcesz włączyć, i kliknij przycisk **OK**.

Pojęcia pokrewne:

- “Tabele buforowane” na stronie 179

Zadania pokrewne:

- “Dodawanie zmaterializowanej tabeli zapytania do tabeli buforowanej” na stronie 183

Dodawanie zmaterializowanej tabeli zapytania do tabeli buforowanej

Po utworzeniu tabeli buforowanej serwer stowarzyszony przechowuje lokalnie dane ze źródła danych w zmaterializowanej tabeli zapytania. Kryteria podane w Kreatorze tabeli buforowanej określają dane, które mają być przechowywane w zmaterializowanej tabeli zapytania.

Aby przechowywać inne dane z tego samego źródła danych, można utworzyć dodatkowe zmaterializowane tabele zapytania dla tej samej tabeli buforowanej.

Wymagania wstępne:

Tabela buforowana dla danego źródła danych musi istnieć.

Procedura:

Aby dodać zmaterializowaną tabelę zapytania do tabeli buforowanej:

1. W Centrum sterowania DB2 rozwiń w drzewie obiektów folder **Buforuj obiekty**, a następnie folder **Tabele buforowane**.
2. Kliknij prawym przyciskiem myszy tabelę buforowaną, do której chcesz dodać zmaterializowaną tabelę zapytania, i kliknij polecenie **Właściwości**. Zostanie otwarte okno Właściwości tabeli buforowanej.
3. Kliknij przycisk **Dodaj**. Zostanie otwarty Kreator tabeli buforowanej.
4. Wykonaj instrukcje kreatora.

Pojęcia pokrewne:

- “Tabele buforowane” na stronie 179
- “Zmaterializowane tabele zapytań i systemy stowarzyszone – przegląd” na stronie 173

Zadania pokrewne:

- “Włączanie pamięci podręcznej” na stronie 182
- “Usuwanie zmaterializowanej tabeli zapytania z tabeli buforowanej” na stronie 183

Usuwanie zmaterializowanej tabeli zapytania z tabeli buforowanej

Gdy nie ma już potrzeby lokalnego przechowywania danych w zmaterializowanej tabeli zapytania, można usunąć tę tabelę z tabeli buforowanej. Jeśli tabela buforowana ma tylko jedną zmaterializowaną tabelę zapytania, jej usunięcie spowoduje usunięcie tabeli buforowanej.

Ograniczenia:

Do usunięcia zmaterializowanej tabeli zapytania z tabeli buforowanej należy użyć Centrum sterowania DB2, aby zapewnić całkowite usunięcie tej tabeli z systemu.

Procedura:

Aby usunąć zmaterializowaną tabelę zapytania z tabeli buforowanej:

1. W Centrum sterowania DB2 rozwiń w drzewie obiektów folder **Buforuj obiekty**, a następnie folder **Tabele buforowane**.

2. Kliknij prawym przyciskiem myszy tabelę buforowaną, z której chcesz usunąć zmaterializowaną tabelę zapytania, i kliknij polecenie **Właściwości**. Zostanie otwarte okno Właściwości tabeli buforowanej.
3. Zaznacz zmaterializowaną tabelę zapytania, którą chcesz usunąć, i kliknij przycisk **Usuń**.

Zadania pokrewne:

- “Dodawanie zmaterializowanej tabeli zapytania do tabeli buforowanej” na stronie 183
- “Usuwanie tabeli buforowanej” na stronie 184

Usuwanie tabeli buforowanej

Gdy nie ma już potrzeby lokalnego przechowywania danych w tabeli buforowanej, można usunąć tę tabelę.

Usunięcie tabeli buforowanej spowoduje następujące skutki:

- Zmaterializowane tabele zapytania utworzone dla tabeli buforowanej zostaną usunięte.
- Harmonogram replikacji między źródłami danych i zmaterializowanymi tabelami zapytań zostanie usunięty.
- Pseudonim dla źródła danych zostanie usunięty, o ile został on utworzony podczas tworzenia tabeli buforowanej. Jeśli użyto wtedy istniejącego pseudonimu, serwer stowarzyszony nie usunie go.

Procedura:

Aby usunąć tabelę buforowaną:

1. W Centrum sterowania DB2 rozwiń w drzewie obiektów folder **Buforuj obiekty**, a następnie folder **Tabele buforowane**.
2. Kliknij prawym przyciskiem myszy tabelę buforowaną, którą chcesz usunąć, i kliknij polecenie **Usuń**.

Zadania pokrewne:

- “Tworzenie tabeli buforowanej” na stronie 181
- “Usuwanie zmaterializowanej tabeli zapytania z tabeli buforowanej” na stronie 183

Rozdział 15. Ograniczenia informacyjne na pseudonimach w systemie stowarzyszonym

W tym rozdziale opisano sposób użycia ograniczeń informacyjnych dla pseudonimów.

Ten rozdział zawiera:

- “Ograniczenia informacyjne pseudonimów”
- “Określanie ograniczeń informacyjnych dla pseudonimów”
- “Określanie ograniczeń informacyjnych dla pseudonimów - przykłady” na stronie 186

Ograniczenia informacyjne pseudonimów

Ograniczenia informacyjne to reguły, których może użyć optymalizator w celu poprawienia wydajności, ale które nie są wymuszane przez menedżera bazy danych. Z ograniczeń informacyjnych można korzystać w odniesieniu do pseudonimów w celu poprawienia wydajności zapytań kierowanych do zdalnych źródeł danych.

Dla pseudonimów można określić następujące typy ograniczeń informacyjnych:

- ograniczenia referencyjne,
- ograniczenia sprawdzające,
- ograniczenia zależności funkcjonalnej,
- ograniczenia klucza podstawowego,
- ograniczenia przez unikalność.

Pojęcia pokrewne:

- “Constraints” w podręczniku *SQL Reference, Volume 1*

Zadania pokrewne:

- “Określanie ograniczeń informacyjnych dla pseudonimów” na stronie 185

Informacje pokrewne:

- “Określanie ograniczeń informacyjnych dla pseudonimów - przykłady” na stronie 186

Określanie ograniczeń informacyjnych dla pseudonimów

Aby poprawić wydajność niektórych zapytań kierowanych do zdalnych źródeł danych, można nałożyć na pseudonimy ograniczenia informacyjne. Jednak serwer stowarzyszony nie wymusza tych ograniczeń ani ich nie sprawdza.

W przypadku relacyjnych źródeł danych ograniczenia informacyjne określa się podczas modyfikacji pseudonimu.

W przypadku nierelacyjnych źródeł danych ograniczenia informacyjne określa się podczas tworzenia lub modyfikacji pseudonimu.

Ograniczenia:

Po zdefiniowaniu ograniczeń informacyjnych dla pseudonimu nie można zmieniać nazw kolumn tego pseudonimu, dopóki nie usunie się ograniczeń.

Procedura:

Ograniczenia informacyjne dla pseudonimu można określać z Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Zaznacz folder **Pseudonimy**:

- Jeśli tworzysz pseudonim, w panelu szczegółów obiektu Centrum sterowania DB2 kliknij pozycję **Utwórz pseudonimy** na liście działań. Zostanie otwarty Kreator tworzenia pseudonimów.

W oknie Utwórz pseudonimy otwórz notatnik Dodaj pseudonim albo notatnik Właściwości dla danego pseudonimu:

- Jeśli planujesz utworzyć jeden pseudonim, kliknij przycisk **Dodaj**. Zostanie otwarty notatnik Dodaj pseudonim.
 - Jeśli do wygenerowania listy pseudonimów użyto opcji wykrywania, wybierz pseudonim, do którego chcesz dodać ograniczenia informacyjne. Następnie kliknij opcję **Właściwości**. Zostanie otwarty notatnik Właściwości.
 - Jeśli modyfikujesz pseudonim, kliknij pseudonim, który chcesz zmodyfikować. W panelu szczegółów obiektu Centrum sterowania DB2 kliknij pozycję **Zmień** na liście działań. Zostanie otwarty notatnik Zmodyfikuj pseudonim.
2. Na stronie Klucze skonfiguruj ograniczenia spójności referencyjnej dla pseudonimu. Możesz także określić ograniczenie klucza podstawowego, klucza unikalnego lub klucza obcego.
 3. Na stronie Ograniczenia sprawdzające określ ograniczenia sprawdzające lub ograniczenia zależności funkcjonalnej dla pseudonimu.
 4. Kliknij przycisk **OK**, aby ustawić ograniczenia informacyjne i zamknąć notatnik.

Aby wykonać to zadanie z wiersza komend DB2:

Uruchom instrukcję CREATE NICKNAME lub ALTER NICKNAME z ustawionymi odpowiednimi atrybutami ograniczenia.

Pojęcia pokrewne:

- “Primary keys” w podręczniku *Administration Guide: Planning*
- “Ograniczenia informacyjne pseudonimów” na stronie 185

Informacje pokrewne:

- “Określanie ograniczeń informacyjnych dla pseudonimów - przykłady” na stronie 186

Określanie ograniczeń informacyjnych dla pseudonimów - przykłady

Poniższe przykłady ilustrują użycie ograniczeń informacyjnych w odniesieniu do pseudonimów.

Informacyjne ograniczenie sprawdzające

W poniższej tabeli zdalnej dane w kolumnie salary (zarobki) są zawsze większe od 10000.

```

| CREATE TABLE account.salary (
|     empno INTEGER NOT NULL PRIMARY KEY,
|     salary INTEGER NOT NULL
| );

```

Utwórz pseudonim dla tej tabeli:

```

| CREATE NICKNAME account.salary FOR myserv.account.salary;

```

Następnie dodaj informacyjne ograniczenie sprawdzające dla pseudonimu, uruchamiając następującą instrukcję:

```

| ALTER NICKNAME account.salary ADD CONSTRAINT cons1 CHECK( salary > 10000 )
| NOT ENFORCED
| ENABLE QUERY OPTIMIZATION;

```

Informacyjne ograniczenie referencyjne: pseudonim z pseudonimem

W tym przykładzie występują dwa pseudonimy: N1 i N2. Kolumna F1 pseudonimu N2 zawiera wartości klucza z kolumny P1 pseudonimu N1. Aby zdefiniować ograniczenie referencyjne dla pseudonimu N2, uruchom następującą instrukcję:

```

| ALTER NICKNAME SCHEMA1.N2 ADD CONSTRAINT ref1
|     FOREIGN KEY (F1) REFERENCES SCHEMA1.N1 (P1)
|     NOT ENFORCED;

```

Informacyjne ograniczenie referencyjne: pseudonim z tabelą

W tym przykładzie kolumna F1 pseudonimu N3 zawiera wartości klucza z kolumny P1 tabeli T1. Aby zdefiniować ograniczenie referencyjne dla pseudonimu N3, uruchom następującą instrukcję:

```

| ALTER NICKNAME SCHEMA1.N3 ADD CONSTRAINT ref1
|     FOREIGN KEY (F1) REFERENCES SCHEMA1.T1 (P1)
|     NOT ENFORCED;

```

Informacyjne ograniczenie referencyjne: tabela z pseudonimem

W tym przykładzie kolumna F1 tabeli T2 zawiera wartości klucza z kolumny P1 pseudonimu N4. Aby zdefiniować ograniczenie referencyjne dla tabeli T2, uruchom następującą instrukcję:

```

| ALTER TABLE SCHEMA1.T2 ADD CONSTRAINT ref1
|     FOREIGN KEY (F1) REFERENCES SCHEMA1.N4 (P1)
|     NOT ENFORCED;

```

Zależność funkcjonalna

W tym przykładzie para wartości z kolumn C1 i C2 w sposób jednoznaczny określa wartość w kolumnie P1. Aby zdefiniować zależność funkcjonalną, uruchom następującą instrukcję:

```

| ALTER NICKNAME SCHEMA1.NICK1 ADD CONSTRAINT FD1 CHECK( P1 DETERMINED BY (C1,C2) )
| NOT ENFORCED ENABLE QUERY OPTIMIZATION;

```

Plik o strukturze tabeli

W tym przykładzie przedstawiono sposób definiowania klucza podstawowego dla pliku o strukturze tabeli.

```

| CREATE NICKNAME MY_FILE (
|     X INTEGER NOT NULL,
|     Y INTEGER,
|     PRIMARY KEY (X) NOT ENFORCED
| ) FOR SERVER MY_SERVER OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT');

```

Schemat gwiazdzisty

Poniższy przykład przedstawia cztery tabele wymiarów i jedną tabelę faktów.

```
CREATE TABLE SCHEMA.FACT (  
    LOCATION_CODE INTEGER NOT NULL,  
    PRODUCT_CODE  INTEGER NOT NULL,  
    CUSTOMER_CODE INTEGER NOT NULL,  
    SDATE         DATE NOT NULL,  
    SALES         INTEGER NOT NULL  
);  
  
CREATE TABLE SCHEMA.LOCATION (  
    LOCATION_CODE INTEGER NOT NULL PRIMARY KEY,  
    STATE         CHAR(2) NOT NULL,  
    SHOP_ID      INTEGER NOT NULL,  
    ...  
);  
  
CREATE TABLE SCHEMA.PRODUCT (  
    PRODUCT_CODE INTEGER NOT NULL PRIMARY KEY,  
    PRODUCT_CAT  INTEGER NOT NULL,  
    PRODUCT_NAME VARCHAR(20) NOT NULL,  
    ...  
);  
  
CREATE TABLE SCHEMA.CUSTOMER (  
    CUSTOMER_CODE INTEGER NOT NULL PRIMARY KEY,  
    NAME          VARCHAR(20) NOT NULL,  
    TEL          VARCHAR(10) NOT NULL,  
    ...  
);  
  
CREATE TABLE SCHEMA.TIMEDIM (  
    SDATE         DATE NOT NULL UNIQUE,  
    YEAR          INTEGER NOT NULL,  
    QUARTER       INTEGER NOT NULL,  
    ...  
);
```

Serwer stowarzyszony tworzy następujące pseudonimy dla tabeli faktów i czterech tabel wymiarów:

```
CREATE NICKNAME SCHEMA.FACT FOR SERVER.SCHEMA.FACT;  
CREATE NICKNAME SCHEMA.LOCATION FOR SERVER.SCHEMA.LOCATION;  
CREATE NICKNAME SCHEMA.PRODUCT FOR SERVER.SCHEMA.PRODUCT;  
CREATE NICKNAME SCHEMA.CUSTOMER FOR SERVER.SCHEMA.CUSTOMER;  
CREATE NICKNAME SCHEMA.TIMEDIM FOR SERVER.SCHEMA.TIMEDIM;
```

Można zdefiniować następujący związek kluczy obcych:

```
ALTER NICKNAME SCHEMA.FACT ADD CONSTRAINT L1 FOREIGN KEY (LOCATION_CODE)  
    REFERENCES SCHEMA.LOCATION(LOCATION_CODE)  
    NOT ENFORCED ENABLE QUERY OPTIMIZATION;  
  
ALTER NICKNAME SCHEMA.FACT ADD CONSTRAINT P1 FOREIGN KEY (PRODUCT_CODE)  
    REFERENCES SCHEMA.PRODUCT(PRODUCT_CODE)  
    NOT ENFORCED ENABLE QUERY OPTIMIZATION;  
  
ALTER NICKNAME SCHEMA.FACT ADD CONSTRAINT C1 FOREIGN KEY (CUSTOMER_CODE)  
    REFERENCES SCHEMA.CUSTOMER(CUSTOMER_CODE)  
    NOT ENFORCED ENABLE QUERY OPTIMIZATION;  
  
ALTER NICKNAME SCHEMA.FACT ADD CONSTRAINT S1 FOREIGN KEY (SDATE)  
    REFERENCES SCHEMA.TIMEDIM(SDATE)  
    NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```


| Gdy wartość kolumny TEL w pseudonimie CUSTOMER jest unikalna, można dodać
| następujące informacyjne ograniczenie przez unikalność:

```
| ALTER NICKNAME SCHEMA.CUSTOMER ADD CONSTRAINT U1 UNIQUE( TEL )  
| NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```

| Gdy wartość kolumny SHOP_ID w pseudonimie LOCATION jednoznacznie określa wartość
| w kolumnie LOCATION_ID, można zdefiniować następującą zależność funkcjonalną:

```
| ALTER NICKNAME SCHEMA.LOCATION  
| ADD CONSTRAINT F1 CHECK( LOCATION_ID DETERMINED BY SHOP_ID )  
| NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```

| Ponieważ wartość kolumny QUARTER w pseudonimie TIMEDIM należy do zakresu od 1 do
| 4, można zdefiniować następujące informacyjne ograniczenie sprawdzające:

```
| ALTER NICKNAME SCHEMA.TIMEDIM  
| ADD CONSTRAINT Q1 CHECK( QUARTER BETWEEN 1 AND 4 )  
| NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```

| Instrukcje w tym przykładzie tworzą pseudonimy dla tabel zdalnych. Gdy tabele zdalne mają
| klucze podstawowe, pseudonimy także mają klucze podstawowe. Pseudonimy tworzone dla
| widoków nie mają kluczy podstawowych. Można wtedy zmienić pseudonim, dodając
| informacyjne ograniczenie klucza głównego. Na przykład:

```
| CREATE NICKNAME SCHEMA.LOCATION FOR SERVER.SH.V_LOCATION;  
| ALTER NICKNAME SCHEMA.LOCATION  
| ADD CONSTRAINT P1 PRIMARY KEY ( LOCATION_CODE ) NOT ENFORCED;
```

| **Zadania pokrewne:**

- | • “Określanie ograniczeń informacyjnych dla pseudonimów” na stronie 185

Rozdział 16. Statystyka pseudonimu

W tym rozdziale opisano sposób pobierania statystyk pseudonimów.

Ten rozdział zawiera:

- “Narzędzie do aktualizacji statystyk pseudonimu - przegląd”
- “Pobieranie statystyk pseudonimów” na stronie 192
- “Pobieranie statystyk pseudonimów z wiersza komend - przykłady” na stronie 194
- “Tworzenie katalogu narzędzi DB2” na stronie 194
- “Wyświetlanie statusu aktualizacji statystyk pseudonimu” na stronie 195

Narzędzie do aktualizacji statystyk pseudonimu - przegląd

Po zarejestrowaniu pseudonimu dla obiektu źródła danych serwer stowarzyszony wprowadza informacje o tym obiekcie do katalogu systemowego w stowarzyszonej bazie danych. Optymalizator zapytań DB2[®] używa tych informacji do planowania pobierania danych z obiektu źródła danych. Stowarzyszona baza danych nie wykrywa automatycznie zmian w obiektach źródła danych, dlatego informacje w katalogu systemowym mogą stać się nieaktualne. Pobieranie statystyk dla pseudonimu zapewni, że optymalizator zapytań będzie korzystał z najnowszych informacji o pseudonimie podczas generowania planów dostępu dla zapytań.

Ze zdalnego źródła danych można pobierać dostępne statystyki pseudonimów bazy danych, kolumn i indeksów.

Można pobierać statystyki pojedynczego pseudonimu lub wszystkich pseudonimów w schemacie DB2 dla określonej definicji serwera DB2. Jeśli część operacji pobierania nie powiedzie się, baza danych wycofa wszystkie zmiany.

W przypadku relacyjnych źródeł danych i źródeł danych ODBC, dla których ustawienia statystyk są inne niż domyślne, można pobierać następujące statystyki, o ile są one dostępne w zdalnym źródle danych:

- CARD
- FPAGES
- NPAGES
- OVERFLOW
- COLCARD
- HIGH2KEY
- LOW2KEY
- NLEAF
- NLEVELS
- CLUSTERFACTOR
- CLUSTERRATIO
- FULLKEYCARD
- FIRSTKEYCARD

W przypadku nierelacyjnych źródeł danych i źródeł danych ODBC, dla których statystyki mają ustawienia domyślne, można pobierać następujące statystyki, o ile są one dostępne w zdalnym źródle danych:

- CARD
- COLCARD
- HIGH2KEY
- LOW2KEY
- FULLKEYCARD
- FIRSTKEYCARD

Statystyki pseudonimu można pobierać przy użyciu Centrum sterowania DB2 lub z wiersza komend programu DB2.

Pobieranie statystyk pseudonimów jest możliwe dla następujących źródeł danych:

- BioRS
- produkty z rodziny DB2 (DRDA)
- Documentum (z wyjątkiem pseudokolumn)
- Informix®
- Microsoft® SQL Server
- ODBC
- Oracle (NET8)
- Sybase (CTLIB)
- Pliki o strukturze tabeli
- Teradata
- XML (tylko dla pseudonimu elementu głównego)

Możliwe jest pobieranie statystyk na serwery stowarzyszone, na których działają systemy operacyjne Microsoft Windows® 2000 i Windows NT®, AIX®, Solaris, HP-UX i Linux.

Zadania pokrewne:

- “Pobieranie statystyk pseudonimów” na stronie 192
- “Wyświetlanie statusu aktualizacji statystyk pseudonimu” na stronie 195

Informacje pokrewne:

- “Pobieranie statystyk pseudonimów z wiersza komend - przykłady” na stronie 194
- Rozdział 32, “Procedura zapisana w bazie SYSPROC.NNSTAT”, na stronie 309

Pobieranie statystyk pseudonimów

Pobieranie statystyk dla pseudonimu zapewnia, że optymalizator zapytań używa informacji o pseudonimie dostępnych obecnie w źródle danych. Aktualne informacje statystyczne o pseudonimie pozwalają poprawić wydajność zapytań. Jednak statystyki są tylko tak dokładne i aktualne, jak aktualne są informacje w zdalnym źródle.

Wymagania wstępne:

Aby można było używać wiersza komend do aktualizacji statystyk, muszą być spełnione następujące wymagania wstępne:

- Serwer stowarzyszony utworzy plik protokołu na serwerze. Katalogi podane w ścieżce muszą istnieć.

- W systemie Windows w ścieżce do protokołu należy użyć dwóch ukośników odwrotnych.
Na przykład: c:\\temp\\nnstat.log.

Ograniczenia:

Jeśli lokalna nazwa lub typ kolumny są inne niż zdalna nazwa lub typ kolumny, program narzędziowy do aktualizacji statystyk pseudonimów nie będzie pobierał statystyk kolumn.

Identyfikator użytkownika używany do połączenia ze stowarzyszoną bazą danych musi być odwzorowany na identyfikator użytkownika tabeli w zdalnym źródle danych.

Procedura:

Statystyki pseudonimu można aktualizować przy użyciu Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Zaznacz pseudonimy, dla których chcesz pobrać bieżące statystyki:

- Aby zaktualizować statystyki dla wszystkich pseudonimów skojarzonych z definicją serwera:
 - a. Rozwiń folder **Obiekty stowarzyszonej bazy danych**.
 - b. Rozwiń folder **Definicje serwerów**, zawierający pseudonimy, które chcesz zaktualizować.
 - c. Kliknij prawym przyciskiem myszy definicję serwera i wybierz polecenie **Statystyka**.
 - d. Wybierz polecenie **Aktualizuj**. Zostanie otwarty notatnik Aktualizuj statystykę.
- Aby zaktualizować statystyki dla pojedynczego pseudonimu, rozwiń folder **Pseudonimy**.
 - a. Kliknij prawym przyciskiem myszy pseudonim, który chcesz zaktualizować, i wybierz polecenie **Statystyka**.
 - b. Wybierz polecenie **Aktualizuj**. Zostanie otwarty notatnik Aktualizuj statystykę.

2. Jeśli katalog narzędzi programu DB2 nie istnieje, zostanie wyświetlone okno, w którym można będzie utworzyć ten katalog.

3. Podaj ustawienia aktualizacji:

- Aby zaktualizować statystyki dla wszystkich pseudonimów skojarzonych z definicją serwera:
 - a. Na stronie Pseudonimy:
 - Wybierz schemat skojarzony z pseudonimami, dla których chcesz zaktualizować statystyki. Jeśli nie wybierzesz schematu, serwer stowarzyszony zaktualizuje statystyki dla wszystkich pseudonimów skojarzonych ze schematem lokalnym.
 - Wybierz istniejący plik protokołu lub wpisz pełną ścieżkę do nowego pliku protokołu.
 - b. Na stronie Harmonogram określ odpowiedni czas uruchomienia aktualizacji statystyk.
- Aby zaktualizować statystykę dla pojedynczego pseudonimu, określ czas uruchomienia aktualizacji statystyki.

Aby wykonać to zadanie z wiersza komend lub w ramach aplikacji, wywołaj procedurę zapisaną w bazie SYSPROC.NNSTAT.

Pojęcia pokrewne:

- “Narzędzie do aktualizacji statystyk pseudonimu - przegląd” na stronie 191

Zadania pokrewne:

- “Wyświetlanie statusu aktualizacji statystyk pseudonimu” na stronie 195

Informacje pokrewne:

- “Pobieranie statystyk pseudonimów z wiersza komend - przykłady” na stronie 194
- Rozdział 32, “Procedura zapisana w bazie SYSPROC.NNSTAT”, na stronie 309

Pobieranie statystyk pseudonimów z wiersza komend - przykłady

W tym przykładzie serwer stowarzyszony pobiera statystyki dla wszystkich pseudonimów na serwerze DB2SERV i nie tworzy protokołu.

```
CALL SYSPROC.NNSTAT('DB2SERV', 'NULL', 'NULL', 'NULL', ?, ?)
```

W tym przykładzie serwer stowarzyszony pobiera statystyki dla pseudonimu STAFF w schemacie ADMIN. Serwer stowarzyszony zapisuje protokół w pliku /home/iiuser/reportlogs/log1.txt.

```
CALL SYSPROC.NNSTAT('NULL', 'ADMIN', 'STAFF', '/home/iiuser/reportlogs/log1.txt', ?, ?)
```

W tym przykładzie serwer stowarzyszony pobiera statystyki dla wszystkich pseudonimów na serwerze DB2Serv w schemacie admin. Serwer stowarzyszony zapisuje protokół w pliku /home/iiuser/stats/recent.log.

```
CALL SYSPROC.NNSTAT('DB2Serv', 'admin', 'NULL', '/home/iiuser/stats/recent.log', ?, ?)
```

Pojęcia pokrewne:

- “Narzędzie do aktualizacji statystyk pseudonimu - przegląd” na stronie 191

Zadania pokrewne:

- “Pobieranie statystyk pseudonimów” na stronie 192

Tworzenie katalogu narzędzi DB2

W celu zaktualizowania statystyki pseudonimu można użyć katalogu narzędzi DB2, aby zaplanować aktualizację. Czynność tę można wykonać wyłącznie w Centrum sterowania DB2. Jeśli w systemie nie ma katalogu narzędzi DB2, zostanie wyświetlony monit o jego utworzenie.

Wymagania wstępne:

Musi być zainstalowany Serwer administracyjny DB2.

Procedura:

Katalog narzędzi DB2 można utworzyć z Centrum sterowania DB2 lub z wiersza komend.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Okno Aktualizuj statystykę jest wyświetlane podczas aktualizacji statystyki pseudonimu
2. Wybierz system, w którym chcesz utworzyć bazę danych dla katalogu narzędzi DB2.

Baza danych musi znajdować się w systemie wpisanym do katalogu, w którym nie przechowuje się obecnie metadanych. Jeśli dany system nie jest wpisany do katalogu, należy go wpisać przed utworzeniem bazy danych dla katalogu narzędzi DB2.

Zadania pokrewne:

- “Pobieranie statystyk pseudonimów” na stronie 192
- “Tools catalog database and DAS scheduler setup and configuration” w podręczniku *Administration Guide: Implementation*

Informacje pokrewne:

- “CREATE TOOLS CATALOG Command” w podręczniku *Command Reference*

Wyświetlanie statusu aktualizacji statystyk pseudonimu

Po zażądaniu aktualizacji statystyk pseudonimu można wyświetlić status aktualizacji. Na podstawie tych informacji można ocenić wydajność zapytań.

Procedura:

Status aktualizacji statystyk pseudonimu można wyświetlić w Centrum sterowania DB2 lub z wiersza komend programu DB2.

Aby wykonać to zadanie w Centrum sterowania DB2:

1. Zaznacz pseudonimy, dla których chcesz wyświetlić status aktualizacji:
 - Aby wyświetlić status aktualizacji dla wszystkich pseudonimów skojarzonych z definicją serwera:
 - a. Rozwiń folder **Obiekty stowarzyszonej bazy danych**.
 - b. Rozwiń folder **Definicje serwerów**, zawierający pseudonimy, które chcesz zaktualizować.
 - c. Kliknij prawym przyciskiem myszy definicję serwera i wybierz polecenie **Statystyka**.
 - Aby wyświetlić status aktualizacji dla konkretnego pseudonimu, rozwiń folder **Pseudonimy**.
 - a. Wybierz pseudonim, który chcesz zaktualizować.
 - b. Kliknij pseudonim prawym przyciskiem myszy i wybierz polecenie **Statystyka**.
2. Wybierz polecenie **Wyświetl wyniki**. Zostanie otwarte okno Statystyka: Wyświetl wyniki.

Aby wykonać to zadanie z wiersza komend, przejrzyj historię aktualizacji w tabeli SYSPROC.FED_STATS.

Pojęcia pokrewne:

- “Narzędzie do aktualizacji statystyk pseudonimu - przegląd” na stronie 191

Zadania pokrewne:

- “Pobieranie statystyk pseudonimów” na stronie 192

Informacje pokrewne:

- “Pobieranie statystyk pseudonimów z wiersza komend - przykłady” na stronie 194

Część 4. Programowanie aplikacji

Rozdział 17. Scenariusz programowania aplikacji

Z poniższych publikacji można się dowiedzieć, w jaki sposób projektować aplikacje przy użyciu programu DB2 Information Integrator i produktu IBM Websphere Portal:

- *IBM Websphere Portal and DB2 Information Integrator*
(<http://www.redbooks.ibm.com/redpieces/pdfs/sg246433.pdf>)
- Sample code from the scenario (<ftp://www.redbooks.ibm.com/redbooks/SG246433/>)

Rozdział 18. Programowanie aplikacji dla systemów stowarzyszonych

W tym rozdziale omówiono zagadnienia, które programiści muszą wziąć pod uwagę podczas tworzenia aplikacji dla systemów stowarzyszonych.

Szczegółowe informacje dotyczące programowania aplikacji zawierają następujące podręczniki:

- *IBM DB2 Universal Database Application Development Guide: Building and Running Applications Version 8*
- *IBM DB2 Universal Database Application Development Guide: Programming Client Applications Version 8*

Jak aplikacje klienckie współpracują ze źródłami danych

Od strony aplikacji klienckich źródła danych w systemie stowarzyszonym widoczne są jako jedna, zbiorcza baza danych. W celu pobrania danych ze źródeł danych aplikacje wprowadzają zapytania SQL do stowarzyszonej bazy danych DB2®. Następnie program DB2 UDB rozprawdza zapytania do odpowiednich źródeł danych i albo zwraca te dane do aplikacji, albo wykonuje działanie, którego dotyczy zapytanie. Stowarzyszona baza danych może łączyć dane z tabel lokalnych i ze zdalnych źródeł danych, tak jak gdyby wszystkie dane rezydowały lokalnie. Na przykład w ramach pojedynczej instrukcji SQL można połączyć dane z lokalnej tabeli programu DB2 for Windows® z danymi z tabeli Informix® i z widoku Sybase. Przetwarzając instrukcje SQL tak, jak gdyby źródła danych były zwykłymi tabelami relacyjnymi lub widokami w stowarzyszonej bazie danych, system stowarzyszony może łączyć dane systemów relacyjnych z danymi w formatach nierelacyjnych.

W systemie stowarzyszonym odwołania do źródeł danych realizowane są za pośrednictwem pseudonimów. *Pseudonim* jest to identyfikator, z którego korzysta aplikacja w celu odwołania się do obiektu źródła danych, na przykład tabeli lub widoku. Dokonując zapisu w źródle danych, na przykład aktualizując zdalną tabelę, aplikacja w programie DB2 może korzystać z instrukcji SQL operujących na pseudonimach. Aplikacja może też korzystać z dialektu SQL źródła danych (bez pseudonimów) w ramach specjalnego typu sesji nazywanej *sesją tranzytową*, która daje bezpośredni dostęp do źródła danych.

Aplikacje korzystające z pseudonimów i dialektu DB2 SQL mają dostęp do wszystkich typów danych rozpoznawanych przez program DB2 UDB.

Katalog systemowy stowarzyszonej bazy danych zawiera informacje dotyczące obiektów w stowarzyszonej bazie danych oraz obiektów w źródłach danych. Ponieważ katalog zawiera informacje na temat całego systemu stowarzyszonego, nazywany jest on *katalogiem globalnym*.

Pojęcia pokrewne:

- “Bezpośrednie kierowanie zapytań do źródeł danych w sesji tranzytovej” na stronie 213

Zadania pokrewne:

- “Odwoływanie się do obiektów źródeł danych za pośrednictwem pseudonimów w instrukcjach SQL” na stronie 202
- “Pseudonimy w instrukcjach DDL” na stronie 202

- “Wpływ statystyk źródła danych na aplikacje” na stronie 203
- “Pseudonimy wywołujące procedury zapisane w bazie” na stronie 205
- “Definiowanie opcji kolumn dla pseudonimów” na stronie 205
- “Dostęp do źródeł danych przy użyciu sesji tranzytowych” na stronie 107

Praca z pseudonimami w aplikacjach

Odwoływanie się do obiektów źródeł danych za pośrednictwem pseudonimów w instrukcjach SQL

W systemie stowarzyszonym nie ma potrzeby definiowania w instrukcjach SQL serwera źródła danych, schematu i obiektu. Zamiast tego w instrukcjach SQL działających na obiektach źródła danych używane są pseudonimy tych obiektów.

Każdy obiekt źródła danych, aby mógł występować w zapytaniu, musi mieć zarejestrowany pseudonim w stowarzyszonej bazie danych.

Używanie pseudonimów w instrukcjach SELECT, INSERT, UPDATE i DELETE:

Załóżmy, że zdefiniowany został pseudonim NFXDEPT mający reprezentować tabelę Informix o nazwie PERSON.DEPT, gdzie:

- PERSON jest schematem źródła danych;
- DEPT jest nazwą tabeli źródła danych.

Instrukcja SELECT * FROM NFXDEPT może być wydawana na serwerze stowarzyszonym. Jednak instrukcja SELECT * FROM PERSON.DEPT nie jest dozwolona (z wyjątkiem sesji tranzytowych). Na serwerze stowarzyszonym nie ma bowiem zarejestrowanego pseudonimu PERSON.DEPT.

Korzystanie z pseudonimów w instrukcjach CREATE TABLE:

Załóżmy, że ma zostać utworzona zmaterializowana tabela zapytania na podstawie tabeli, dla której zdefiniowano pseudonim. Przykładowa instrukcja CREATE TABLE ma postać:

```
CREATE TABLE nazwa_tabeli LIKE pseudonim
```

Pojęcia pokrewne:

- “Bezpośrednie kierowanie zapytań do źródeł danych w sesji tranzytowej” na stronie 213

Zadania pokrewne:

- “Uwagi i ograniczenia dotyczące sesji tranzytowych w systemach stowarzyszonych” na stronie 214
- “Sesje tranzytowe do źródeł danych Oracle” na stronie 214

Pseudonimy w instrukcjach DDL

Każdy obiekt źródła danych, aby mógł wystąpić w instrukcji DDL, musi mieć zarejestrowany pseudonim w stowarzyszonej bazie danych. Oto przykłady instrukcji DDL używanych w systemach stowarzyszonych:

Korzystanie z pseudonimów w instrukcji COMMENT ON:

Instrukcja COMMENT ON dodaje lub zastępuje komentarze w katalogu globalnym stowarzyszonej bazy danych. Instrukcja COMMENT ON jest poprawna dla pseudonimu i dla kolumn zdefiniowanych w pseudonimie. Instrukcja ta nie aktualizuje katalogów źródła danych.

Korzystanie z pseudonimów w instrukcjach GRANT i REVOKE:

Instrukcje GRANT i REVOKE mogą być używane do nadawania niektórych uprawnień do pseudonimów wszystkim użytkownikom i grupom. Jednak program DB2 nie wykonuje odpowiednich instrukcji GRANT lub REVOKE dla obiektu w źródle danych, do którego odwołuje się pseudonim.

Na przykład założmy, że użytkownik JON tworzy pseudonim dla tabeli Oracle pozbawionej indeksu. Pseudonim nosi nazwę ORAREM1. Następnie administrator bazy Oracle definiuje dla tej tabeli indeks. Teraz użytkownik EILEEN chce wprowadzić do stowarzyszonej bazy danych DB2 informacje o utworzonym indeksie, dzięki czemu optymalizator zapytań będzie mógł opracowywać strategię bardziej wydajnego dostępu do tabeli. EILEEN może wprowadzić tę informację do stowarzyszonej bazy danych, tworząc specyfikację indeksu dla pseudonimu ORAREM1.

Informacja o indeksie jest zapisywana w widoku katalogu SYSSTAT.INDEXES. Należy użyć instrukcji GRANT, aby nadać użytkownikowi EILEEN uprawnienie indeksowania tego pseudonimu.

```
GRANT INDEX ON NICKNAME ORAREM1 TO USER EILEEN
```

Aby odebrać uprawnienie umożliwiające EILEEN tworzenie specyfikacji indeksu dla pseudonimu ORAREM1, należy użyć instrukcji REVOKE:

```
REVOKE INDEX ON ORAREM1 FROM USER EILEEN
```

Zadania pokrewne:

- “Tworzenie specyfikacji indeksów dla obiektów źródła danych” na stronie 74
- “Instrukcje SQL używane w odniesieniu do pseudonimów” na stronie 101

Informacje pokrewne:

- “COMMENT statement” w podręczniku *SQL Reference, Volume 2*
- “GRANT (Table, View, or Nickname Privileges) statement” w podręczniku *SQL Reference, Volume 2*
- “REVOKE (Table, View, or Nickname Privileges) statement” w podręczniku *SQL Reference, Volume 2*

Wpływ statystyk źródła danych na aplikacje

Podczas tworzenia pseudonimu dla obiektu źródła danych do globalnego katalogu stowarzyszonej bazy danych wprowadzane są informacje o tym obiekcie. Optymalizator zapytań DB2 używa tych informacji do planowania pobierania danych z obiektu. Należy dbać o aktualność tych danych. Serwer stowarzyszony nie wykrywa automatycznie zmian w obiektach źródła danych.

Statystyka katalogowa zapisana w katalogu globalnym:

Informacje zapisane w katalogu globalnym na temat obiektu źródła danych zależą od typu obiektu. W przypadku tabel i widoków baz danych w katalogu globalnym zapisana jest nazwa obiektu, nazwy kolumn oraz ich atrybuty.

W przypadku tabeli informacje te obejmują także:

- Statystyki. Przykładem może być liczba wierszy i liczba stron, na których istnieją wiersze. Należy dbać o to, aby program DB2 UDB dysponował aktualnymi statystykami. Dlatego przed utworzeniem pseudonimu należy w źródle danych uruchomić dla odpowiedniej tabeli komendę stanowiącą odpowiednik komendy RUNSTATS.
- Opisy indeksów. Jeśli tabela nie ma indeksów, można do katalogu wpisać metadane, które zazwyczaj zawiera definicja indeksu. Na przykład można określić, które kolumny w tabeli mają unikalne wartości oraz czy są w niej unikalne wiersze. Do wygenerowania tych metadanych, nazywanych łącznie *specyfikacją indeksu*, można użyć instrukcji CREATE INDEX w odwołaniu do pseudonimu dla tabeli. W wywołaniu instrukcji CREATE INDEX należy zawrzeć klauzulę SPECIFICATION ONLY, aby utworzyć tylko specyfikację indeksu. Nie ma potrzeby tworzenia indeksu jako takiego.

Aby ustalić, jakie informacje o źródle danych zapisane są w katalogu globalnym, należy uruchomić zapytanie względem widoków katalogowych SYSCAT.TABLES i SYSCAT.COLUMN. Aby ustalić, jakie informacje o indeksach źródła danych zawiera katalog, lub co zawiera konkretna specyfikacja indeksu, należy wykonać zapytanie względem widoku katalogowego SYSCAT.INDEXES.

Modyfikowanie aplikacji tak, aby odwoływały się do widoku SYSSTAT zamiast widoku SYSCAT:

Widok SYSCAT w programie DB2 wersja 8 jest przeznaczony tylko do odczytu. Wszelkie operacje UPDATE i INSERT względem widoku SYSCAT zakończą się niepowodzeniem. Zalecaną metodą modyfikowania katalogu systemowego jest korzystanie z widoków SYSSTAT. Aplikacje odwołujące się do widoku SYSCAT należy zmodyfikować, tak aby odwoływały się do widoku SYSSTAT.

Pojęcia pokrewne:

- “Catalog statistics” w podręczniku *Administration Guide: Performance*
- “Catalog statistics tables” w podręczniku *Administration Guide: Performance*
- “Stowarzyszona baza danych” na stronie 7

Zadania pokrewne:

- “Tworzenie specyfikacji indeksów dla obiektów źródła danych” na stronie 74
- “Cechy pseudonimu wpływające na globalną optymalizację” na stronie 149

Informacje pokrewne:

- “SYSCAT.COLUMNS catalog view” w podręczniku *SQL Reference, Volume 1*
- “SYSCAT.INDEXES catalog view” w podręczniku *SQL Reference, Volume 1*
- “SYSCAT.TABLES catalog view” w podręczniku *SQL Reference, Volume 1*
- Rozdział 19, “Widoki w tabeli katalogu globalnego zawierające informacje o systemie stowarzyszonym”, na stronie 219

Pseudonimy wywołujące procedury zapisane w bazie

Dokonując migracji aplikacji programu DataJoiner, w których za pośrednictwem pseudonimów wywoływane są procedury zapisane w bazie, należy wprowadzić w tych aplikacjach odpowiednie zmiany. Program DB2 Information Integrator nie umożliwi obecnie wywoływania procedur zapisanych w bazie za pośrednictwem pseudonimów.

Definiowanie opcji kolumn dla pseudonimów

Opcje kolumn są parametrami instrukcji CREATE NICKNAME i ALTER NICKNAME. Opcje kolumn można definiować podczas tworzenia pseudonimu lub na drodze modyfikacji pseudonimu już istniejącego.

Informacje o ustawieniach opcji kolumn są przechowywane w katalogu globalnym.

Nierelacyjne źródła danych

Każde opakowanie nierelacyjne ma własny zestaw opcji kolumn. Opcje te są ustawiane zazwyczaj w instrukcji CREATE NICKNAME.

Relacyjne źródła danych

W przypadku relacyjnych źródeł danych istnieją dwie opcje kolumn: NUMERIC_STRING i VARCHAR_NO_TRAILING_BLANKS.

Ustawianie opcji NUMERIC_STRING

Jeśli kolumna z łańcuchami liczbowymi w źródle danych zawiera wyłącznie cyfry i nie zawiera żadnych innych znaków ani spacji, opcji NUMERIC_STRING można nadać ustawienie 'Y'. Umożliwi to optymalizację zapytań dotyczących tej kolumny pod kątem operacji sortowania i porównywania. Na przykład:

```
ALTER NICKNAME pseudonim
  ALTER COLUMN lokalna_nazwa_kolumny
  OPTIONS (SET NUMERIC_STRING 'Y')
```

Ustawianie opcji VARCHAR_NO_TRAILING_BLANKS

Niektóre źródła danych, takie jak Oracle, nie korzystają z tego samego mechanizmu porównywania łańcuchów z dopełnianiem znakami odstępu, który jest używany w programie DB2 dla systemów Linux, UNIX i Windows. Dotyczy to takich typów danych, jak VARCHAR i VARCHAR2. W efekcie predykaty odwołujące się do tych typów danych muszą być zmienione przez optymalizator zapytań, aby zagwarantować spójność wyników zapytań. Modyfikacja instrukcji zapytania może mieć negatywny wpływ na wydajność. Odpowiednie ustawienie tej opcji dla danej kolumny stanowi dla optymalizatora zapytań informację, na podstawie której możliwe będzie wygenerowanie bardziej wydajnego kodu SQL.

Na przykład:

```
ALTER NICKNAME pseudonim
  ALTER COLUMN lokalna_nazwa_kolumny
  OPTIONS (SET VARCHAR_NO_TRAILING_BLANKS 'Y')
```

Informacje pokrewne:

- Rozdział 24, “Opcje kolumn pseudonimu dla systemów stowarzyszonych”, na stronie 253

Tworzenie i używanie widoków stowarzyszonych

Widok w stowarzyszonej bazie danych, którego tabele podstawowe znajdują się w zdalnych źródłach danych, jest nazywany *widokiem stowarzyszonym*. Do tabel podstawowych widok stowarzyszony odwołuje się za pośrednictwem pseudonimów, a nie nazw tabel ze źródła danych.

Zalety korzystania z widoków stowarzyszonych są podobne jak w przypadku widoków zdefiniowanych na podstawie wielu tabel lokalnych w scentralizowanym środowisku menedżera bazy danych:

- widoki przedstawiają dane w sposób zintegrowany;
- z widoku można wykluczyć te kolumny tabel, które zawierają dane poufne lub chronione.

Ograniczenia:

Widoki stowarzyszone utworzone na podstawie wielu obiektów źródeł danych są przeznaczone tylko do odczytu i nie zezwalają na modyfikacje danych.

Widoki stowarzyszone utworzone na podstawie tylko jednego obiektu źródła danych mogą, ale nie muszą, być widokami tylko do odczytu.

- Widok stowarzyszony utworzony na podstawie pojedynczego nierelacyjnego źródła danych jest przeznaczony tylko do odczytu.
- Widok stowarzyszony utworzony na podstawie pojedynczego relacyjnego źródła danych może zezwalać na aktualizacje, zależnie od zawartości instrukcji CREATE VIEW.

Procedura:

Widok stowarzyszony tworzy się na podstawie obiektów źródeł danych, dla których utworzono pseudonimy. Działanie tworzenia w stowarzyszonej bazie danych widoku danych ze źródła danych nazywa się niekiedy “tworzeniem widoku dla pseudonimu”. Odzwierciedla to fakt, że do utworzenia widoku stowarzyszonego klauzula pełnej selekcji w instrukcji CREATE VIEW musi odwoływać się do pseudonimów poszczególnych tabel i widoków, które będą podstawą widoku stowarzyszonego.

Przykład: Tworzenie widoku stowarzyszonego, który łączy podobne dane z kilku obiektów źródła danych:

Załóżmy, że dane klientów pewnej firmy znajdują się na trzech odrębnych serwerach: w Europie, Azji i w Ameryce Południowej. Na serwerze europejskim dane znajdują się w tabeli Oracle. Pseudonim tej tabeli to `ORA_EU_CUST`. Dane klientów azjatyckich przechowywane są w tabeli Sybase. Pseudonim tej tabeli to `SYB_AS_CUST`. Dane klientów południowoamerykańskich są zapisane w tabeli Informix. Pseudonim tej tabeli to `INFMX_SA_CUST`. Każda z tabel zawiera kolumnę z numerem klienta (`CUST_NO`), nazwą klienta (`CUST_NAME`), numerem produktu (`PROD_NO`) i wielkością zamówienia (`QUANTITY`). Składnia instrukcji tworzącej na podstawie tych trzech pseudonimów widok łączący wszystkie dane klientów jest następująca:

```
CREATE VIEW FV1
AS SELECT * FROM ORA_EU_CUST
UNION
SELECT * FROM SYB_AS_CUST
UNION
SELECT * FROM INFMX_SA_CUST
```

Przykład: Łączenie danych w celu utworzenia widoku stowarzyszonego:

Załóżmy, że dane klientów przechowywane są na jednym serwerze, a dane o sprzedaży na drugim. Dane klientów znajdują się w tabeli bazy danych Oracle. Pseudonim tej tabeli to ORA_EU_CUST. Dane dotyczące sprzedaży znajdują się w tabeli bazy danych Sybase. Pseudonim tej tabeli to SYB_SALES. Konieczne jest dopasowanie informacji o klientach do informacji o zakupach dokonanych przez klientów. W każdej z tabel znajduje się kolumna z numerem klienta (CUST_NO). Składnia instrukcji tworzącej na podstawie dwóch pseudonimów widok stowarzyszony łączący te dane jest następująca:

```
CREATE VIEW FV4
AS SELECT A.CUST_NO, A.CUST_NAME, B.PROD_NO, B.QUANTITY
FROM ORA_EU_CUST A, SYB_SALES B
WHERE A.CUST_NO=B.CUST_NO
```

Zadania pokrewne:

- “Uzyskiwanie dostępu do danych heterogenicznych za pośrednictwem widoków stowarzyszonych” na stronie 108

Informacje pokrewne:

- “CREATE VIEW statement” w podręczniku *SQL Reference, Volume 2*

Korzystanie z poziomów odseparowania do podtrzymania spójności danych

Poziom odseparowania skojarzony z procesem aplikacji definiuje stopień, w jakim ten proces jest odseparowany od innych równoległe wykonywanych procesów. Poziom odseparowania określany jest jako atrybut pakietu i ma zastosowanie do aplikacji korzystających z pakietu. Informacja o poziomie odseparowania jest używana podczas przygotowywania lub wiązania aplikacji.

Blokady zakładane są na poziomie wiersza tabeli podstawowej. Jednak menedżer bazy danych może zastąpić szereg blokad wierszy pojedynczą blokadą tabeli. Mówimy wtedy o eskalacji blokady. Proces aplikacji ma gwarancję blokady przynajmniej na minimalnym wnioskowanym poziomie.

Aby utrzymać spójność danych w tabeli w źródle danych, można zażądać, aby wiersze tabeli były blokowane na określonym poziomie odseparowania. Na przykład, aby zapewnić sobie wyłączny dostęp do wiersza, należy dla tego wiersza określić poziom odseparowania odczytu powtarzalnego (RR).

Serwer stowarzyszony odwzorowuje podane poziomy odseparowania na odpowiednie poziomy w źródle danych. Przy każdym połączeniu ze źródłem danych opakowanie ustala poziom odseparowania właściwy dla programu DB2 dla systemów Linux, UNIX i Windows. Następnie ustawiany jest odpowiedni poziom odseparowania w zdalnym źródle danych. W razie braku dokładnego odpowiednika serwer stowarzyszony dobiera najbliższy silniejszy poziom. Po nawiązaniu połączenia ze źródłem danych poziom odseparowania nie może być zmieniony przez cały czas trwania połączenia.

Dostępne poziomy odseparowania, to:

CS	stabilność kursora
RR	odczyt powtarzalny
RS	stabilność odczytu
UR	niezatwierdzony odczyt

Procedura:

W poniższej tabeli zestawiono poziomy odseparowania, które można konfigurować w obsługiwanych źródłach danych.

Tabela 17. Odpowiedniość poziomów odseparowania między serwerem stowarzyszonym a źródłami danych.

Serwer stowarzyszony DB2	stabilność kursora	odczyt powtarzalny	stabilność odczytu	niezatwierdzony odczyt
Rodzina produktów DB2	stabilność kursora	odczyt powtarzalny	stabilność odczytu	niezatwierdzony odczyt
Informix	stabilność kursora	odczyt powtarzalny	odczyt powtarzalny	odczyt brudny
Microsoft SQL Server	odczyt zatwierdzony	z możliwością szeregowania	odczyt powtarzalny	odczyt niezatwierdzony
ODBC	odczyt zatwierdzony	z możliwością szeregowania	odczyt powtarzalny	odczyt niezatwierdzony
Oracle	odczyt zatwierdzony	z możliwością szeregowania	z możliwością szeregowania	odczyt zatwierdzony
Sybase	poziom 1	poziom 3	poziom 3	poziom 0

W źródłach OLE DB, Teradata i nierelacyjnych źródłach danych nie funkcjonuje mechanizm podobny do poziomów odseparowania programu DB2 dla systemów Linux, UNIX i Windows. Nie istnieje w związku z tym odwzorowanie między poziomami odseparowania DB2 a źródłami OLE DB, Teradata i nierelacyjnymi źródłami danych.

Pojęcia pokrewne:

- “Isolation levels” w podręczniku *SQL Reference, Volume 1*
- “Applications in Host or iSeries Environments” w podręczniku *Application Development Guide: Programming Client Applications*
- “Jak aplikacje klienckie współpracują ze źródłami danych” na stronie 201

Stowarzyszona obsługa obiektów LOB

System stowarzyszonej bazy danych umożliwia operowanie dużymi obiektami (LOB) w zdalnych źródłach danych. Ponieważ obiekty LOB mogą mieć bardzo duże rozmiary, przekazywanie ich ze zdalnego źródła danych może być bardzo czasochłonne. Stowarzyszona baza danych DB2[®] zawsze dąży do ograniczenia transferu danych LOB ze źródeł danych do niezbędnego minimum, a także do tego, by objęte zapytaniem dane LOB przekazać bezpośrednio ze źródła danych do aplikacji inicjującej, bez materializowania tych danych na serwerze DB2 UDB.

W systemie stowarzyszonym obsługiwane są operacje SELECT na danych typu LOB w źródłach danych DRDA[®], Informix[®], Microsoft[®] SQL Server, Oracle i Sybase. Na przykład:

```
SELECT empname, picture FROM infmx_emp_table  
WHERE empno = '01192345'
```

Gdzie *picture* jest kolumną typu LOB, a *infmx_emp_table* jest pseudonimem odwołującym się do tabeli Informix zawierającej dane o pracownikach.

System stowarzyszony obsługuje operacje SELECT, INSERT, UPDATE i DELETE na obiektach LOB w źródłach danych Oracle (wersja 8 lub nowsza).

Operacje odczytu i zapisu obsługiwane w programie DB2 dla systemów Linux, UNIX® i Windows® w wersji 8 są wymienione w następującej tabeli:

Tabela 18. Obsługa odczytu i zapisu obiektów LOB

Źródło danych	Typ operacji
Rodzina produktów DB2 ¹	tylko odczyt
BioRS	tylko odczyt
BLAST	odczyt i powiązanie wejściowe
Entrez	tylko odczyt
HMMER	odczyt i powiązanie wejściowe
Informix	tylko odczyt
Microsoft SQL Server	tylko odczyt
Oracle (opakowanie NET8)	odczyt i zapis
ODBC	tylko odczyt
Sybase	tylko odczyt
Teradata	tylko odczyt
Usługi WWW	tylko odczyt i powiązanie wyjściowe tylko dla typu CLOB
WebSphere® Business Integration	tylko odczyt i powiązanie wyjściowe tylko dla typu CLOB
XML	tylko odczyt

Uwagi:

1. Do obsługi obiektów LOB wymagany jest program DB2 UDB for iSeries™ Version 5 (lub nowszy). Program DB2 Information Integrator w wersji 8 nie może uzyskiwać dostępu do danych typu LOB w programie DB2 UDB dla systemów Linux, UNIX i Windows, wersja 7.

Obiekty LOB Teradata

Obiekty LOB w źródłach Teradata różnią się nieco od obiektów LOB programu DB2. Program Teradata nie oferuje żadnych typów danych o takich wielkościach, jak typ LOB w programie DB2 UDB. Jednak niektóre typy danych Teradata mogą mieć długość do 64000 bajtów. Są to typy CHAR, VARCHAR, BYTE, VARBYTE, GRAPHIC i VARGRAPHIC. Wszystkie te typy danych Teradata są odwzorowywane na typy danych DB2 LOB, gdy ich długość przekracza wartość maksymalną odpowiedniego typu danych DB2.

Długości obiektów LOB

Niektóre źródła danych, jak Oracle i Informix, nie przechowują informacji o długości kolumn LOB w katalogach systemowych. Przy tworzeniu pseudonimu dotyczącego tabeli pobierane są informacje z katalogu systemowego źródła danych, łącznie z długościami kolumn. Ponieważ dla kolumn typu LOB długość nie jest określona, w stowarzyszonej bazie danych przyjmowane jest założenie, że kolumna może mieć maksymalną dozwoloną długość dla typu LOB w programie DB2 dla systemów Linux, UNIX i Windows. Dlatego w katalogu stowarzyszonej bazy danych na serwerze DB2 dla systemów Linux, UNIX i Windows długość kolumny dla pseudonimu zostaje zapisana jako długość maksymalna.

Pojęcia pokrewne:

- “Wskaźniki LOB” na stronie 210

- “Ograniczenia dotyczące obiektów LOB” na stronie 210

Informacje pokrewne:

- Rozdział 27, “Domyślne proste odwzorowania typów”, na stronie 269

Stowarzyszona obsługa obiektów LOB - informacje szczegółowe

Wskaźniki LOB

Aplikacje mogą żądać wskaźników LOB dla obiektów LOB przechowywanych w zdalnych źródłach danych. *Wskaźnik LOB* to 4-bajtowa wartość przechowywana w zmiennej języka bazowego. Za pośrednictwem wskaźnika LOB aplikacja odwołuje się do wartości obiektu LOB (lub wyrażenia LOB) przechowywanej w systemie baz danych. Za pomocą wskaźnika LOB aplikacja może manipulować wartością obiektu LOB tak, jak gdyby wartość ta była przechowywana w zwykłej zmiennej języka bazowego. Korzystanie z wskaźników LOB eliminuje potrzebę przenoszenia wartości LOB z serwera źródła danych do aplikacji (i ewentualnie z powrotem).

Program DB2[®] UDB może pobierać obiekty LOB ze zdalnych źródeł danych, zapisywać je na serwerze stowarzyszonym, a następnie tworzyć wskaźniki LOB do przechowywanych obiektów LOB. Wskaźniki LOB są zwalniane, gdy:

- aplikacja użyje instrukcji SQL FREE LOCATOR,
- aplikacja użyje instrukcji COMMIT,
- stowarzyszona instancja bazy danych DB2 zostanie zrestartowana.

Pojęcia pokrewne:

- “Large object locators” w podręczniku *Application Development Guide: Programming Server Applications*
- “Stowarzyszona obsługa obiektów LOB” na stronie 208
- “Ograniczenia dotyczące obiektów LOB” na stronie 210

Ograniczenia dotyczące obiektów LOB

W systemach stowarzyszonych obowiązują następujące ograniczenia dotyczące obiektów LOB:

- stowarzyszona baza danych nie może powiązać zdalnych obiektów LOB ze zmienną referencji do pliku,
- obiekty LOB nie są obsługiwane w sesjach tranzycyjnych.

Pojęcia pokrewne:

- “Stowarzyszona obsługa obiektów LOB” na stronie 208
- “Wskaźniki LOB” na stronie 210

Rozproszone żądania kierujące zapytania do źródeł danych

Zapytania wprowadzane do stowarzyszonej bazy danych mogą żądać wyników z jednego źródła danych, ale najczęściej ich żądania obejmują wiele źródeł danych. Ponieważ typowe zapytanie jest rozproszone na wiele źródeł danych, nazywa się je *żądaniem rozproszonym*. W ogólnym przypadku w żądaniach rozproszonych mogą być stosowane trzy konwencje języka SQL umożliwiające określenie miejsca, z którego mają być pobrane dane: podzapytania, operatory mnogościowe oraz podselekcje łączy.

Założmy, że serwer stowarzyszony został skonfigurowany do korzystania ze źródeł danych DB2 for OS/390, DB2 for iSeries i Oracle. W każdym źródle danych przechowywana jest tabela z informacjami o pracownikach. Serwer stowarzyszony odwołuje się do tych tabel za pomocą pseudonimów wskazujących miejsca, gdzie rezydują tabele.

UDB390_EMPLOYEES

Pseudonim tabeli zawierającej informacje o pracownikach, znajdującej się w źródle danych DB2 for OS/390.

iSERIES_EMPLOYEES

Pseudonim tabeli zawierającej informacje o pracownikach, znajdującej się w źródle danych DB2 for iSeries.

ORA_EMPLOYEES

Pseudonim tabeli zawierającej informacje o pracownikach, znajdującej się w źródle danych Oracle.

ORA_REGIONS

Pseudonim znajdującej się w źródle danych Oracle tabeli zawierającej informacje regionach zamieszkania pracowników.

Poniższe przykłady ilustrują trzy konwencje języka SQL stosowane w żądaniach rozproszonych; użyto w nich pseudonimów zdefiniowanych dla każdej tabeli.

Przykład: żądanie rozproszone z podzapytaniem:

Tabela iSERIES_EMPLOYEES zawiera numery telefonów pracowników mieszkających w Azji. Zawiera ona także kody regionów skojarzonych z tymi numerami telefonów, ale nie zawiera samych regionów reprezentowanych przez te kody. W tabeli ORA_REGIONS znajdują się zarówno kody, jak i regiony. W poniższym zapytaniu użyto podzapytania, aby znaleźć kod regionu dla Chin. Następnie zapytanie to na podstawie kodu regionu zwraca listę pracowników, którzy mają telefony w Chinach.

```
SELECT name, telephone FROM db2admin.iSERIES_EMPLOYEES
WHERE region_code IN
  (SELECT region_code FROM dbadmin.ora_regions
   WHERE region_name = 'CHINY')
```

Przykład: żądanie rozproszone z operatorami mnogościowymi:

Serwer stowarzyszony obsługuje trzy operatory mnogościowe: UNION, EXCEPT i INTERSECT.

- Operator mnogościowy UNION służy do łączenia wierszy spełniających warunki co najmniej jednej z kilku instrukcji SELECT.
- Operator mnogościowy EXCEPT służy do pobierania tych wierszy, które spełniają pierwszą instrukcję SELECT i jednocześnie nie spełniają warunku drugiej instrukcji.
- Operator mnogościowy INTERSECT służy do pobierania tych wierszy, które spełniają warunki obydwu instrukcji SELECT.

Ze wszystkimi trzema operatorami mnogościowymi można używać operandu ALL, aby zaznaczyć, że powtarzające się wiersze nie powinny być usuwane z wyniku operacji. Pozwala to wyeliminować potrzebę dodatkowego sortowania.

Poniższe zapytanie pobiera wszystkie nazwiska pracowników i kody regionów, które są obecne zarówno w tabeli iSERIES_EMPLOYEES, jak i w tabeli UDB390_EMPLOYEES, pomimo tego, że każda tabela rezyduje w innym źródle danych.

```

SELECT name, region_code
  FROM iseries_employees
INTERSECT
SELECT name, region_code
  FROM udb390_employees

```

Przykład: żądanie rozproszone z łączaniem:

Łączenie relacyjne tworzy tabelę wynikową, która zawiera kombinację kolumn pobranych z kilku tabel. Należy określić kryteria ograniczające wielkość wierszy w tabeli wynikowej.

Przedstawione poniżej zapytanie łączy nazwiska pracowników i odpowiadające im kody regionów przez porównanie kodów występujących w obydwu tabelach. Każda tabela rezyduje w innym źródle danych.

```

SELECT t1.name, t2.region_name
  FROM dbadmin.iSERIES_employees t1, dbadmin.ora_regions t2
 WHERE t1.region_code = t2.region_code

```

Optymalizacja żądań rozproszonych przy użyciu opcji serwera

W systemie stowarzyszonym wprowadzenie do katalogu globalnego informacji dotyczących całego źródła danych oraz sterowanie interakcjami programu DB2 UDB ze źródłem danych odbywa się przy użyciu parametrów zwanych *opcjami serwera*. Możliwe jest na przykład:

- Wpisanie do katalogu identyfikatora instancji w wyniku przypisania go jako wartości opcji serwera NODE.
- Użycie opcji serwera VARCHAR_NO_TRAILING_BLANKS w celu poinformowania optymalizatora, że żadna kolumna typu VARCHAR rezydująca na serwerze źródła danych nie zawiera końcowych znaków odstępu; Ta opcja serwera jest przeznaczona tylko dla źródeł danych Oracle. Z opcji tej można korzystać tylko wtedy, gdy ma się pewność, że żadne kolumny typu VARCHAR2 w żadnym obiekcie, do którego odwołuje się pseudonim, nie zawierają końcowych znaków odstępu. W przeciwnym razie należy użyć opcji kolumny w celu określenia obiektów na serwerze, które nie mają końcowych znaków odstępu. Opcja kolumny również nazywa się VARCHAR_NO_TRAILING_BLANKS.
- Przypisywanie opcji serwera PLAN_HINTS wartości, która umożliwi programowi DB2 przekazanie do źródła danych Oracle fragmentów instrukcji zwanych *wskazówkami dotyczącymi planu*. Wskazówki dotyczące planu mogą pomóc optymalizatorowi źródła danych w podjęciu decyzji, którego indeksu użyć podczas dostępu do tabeli, i jaką kolejność łączenia zastosować podczas pobierania danych do tabeli wynikowej.

Opcje serwera dla systemu stowarzyszonego ustawia najczęściej administrator bazy danych. Jednak także programista może użyć opcji serwera, aby wspomóc optymalizację zapytań. Załóżmy na przykład, że dla źródeł danych ORACLE1 i ORACLE2 opcja serwera PLAN_HINTS ma wartość domyślną 'N' (nie: dla źródła danych nie będą przekazywane wskazówki dotyczące planu). Programista napisał żądanie rozproszone, które wybiera dane ze źródeł ORACLE1 i ORACLE2. Wiadomo, że wskazówki planu mogłyby pomóc optymalizatorowi wybrać lepszą strategię dostępu do danych w tych źródłach. Można wtedy przesłonić wartość domyślną opcji wartością 'Y' (tak: wskazówki planu mają być przekazywane) na czas połączenia aplikacji ze stowarzyszoną bazą danych. Po zakończeniu połączenia ze źródłem danych zostanie automatycznie przywrócone ustawienie 'N'.

Procedura:

Do ustawiania lub zmiany opcji serwera służy instrukcja SET SERVER OPTION. Aby mieć pewność, że ustawienie zostanie wprowadzone, należy użyć instrukcji SET SERVER

OPTION bezpośrednio po instrukcji CONNECT. Opcja serwera zostanie ustawiona na czas trwania połączenia ze stowarzyszoną bazą danych.

Zalecenie: Instrukcję należy przygotować dynamicznie. Instrukcja SET SERVER OPTION działa tylko w odniesieniu do dynamicznych instrukcji SQL.

Informacje pokrewne:

- “SET SERVER OPTION statement” w podręczniku *SQL Reference, Volume 2*
- Rozdział 21, “Opcje serwera w systemie stowarzyszonym”, na stronie 225

Korzystanie z sesji tranzytowych w aplikacjach

Sesje tranzytowe umożliwiają aplikacjom bezpośrednie komunikowanie się z serwerem przy użyciu jego rodzimych metod dostępu klienta i rodzimego dialektu języka SQL.

Bezpośrednie kierowanie zapytań do źródeł danych w sesji tranzytowej

Sesje tranzytowe są przydatne w sytuacjach, gdy:

- aplikacje muszą tworzyć obiekty w źródle danych lub wykonywać operacje INSERT, UPDATE lub DELETE,
- program DB2[®] UDB nie obsługuje pewnych szczególnych operacji w źródle danych.

Procedura:

Do inicjowania sesji tranzytowej i uzyskiwania bezpośredniego dostępu do serwera służy instrukcja SET PASSTHRU. Instrukcji tej można używać dynamicznie. Oto przykład zastosowania tej instrukcji:

```
SET PASSTHRU ORACLE1
```

Ta instrukcja SET PASSTHRU rozpoczyna sesję tranzytową do źródła danych korzystającego z serwera ORACLE1. ORACLE1 to nazwa serwera źródła danych zarejestrowana podczas tworzenia definicji serwera.

Po otwarciu sesji tranzytowej w odwołaniach do obiektów należy używać ich prawdziwych nazw, a nie pseudonim. Konieczne jest użycie dialektu języka SQL obsługiwane przez źródło danych, chyba że odwołanie dotyczy źródła danych programu DB2 UDB.

Wprowadzenie w sesji tranzytowej instrukcji statycznej spowoduje przesłanie jej do serwera stowarzyszonego, gdzie zostanie ona przetworzona. W celu wprowadzenia instrukcji SQL, która zostanie przetworzona w źródle danych, należy ją przygotować dynamicznie w sesji tranzytowej i wykonać w czasie trwania sesji. Aby przygotować instrukcje dynamicznie w sesji tranzytowej:

- W przypadku instrukcji SELECT: użyj wraz z nią instrukcji PREPARE, a następnie użyj instrukcji OPEN, FETCH i CLOSE w celu uzyskania dostępu do wyników zapytania.
- W przypadku obsługiwanej instrukcji innej niż SELECT dostępne są dwie opcje. Można użyć instrukcji PREPARE, aby przygotować obsługiwaną instrukcję, a następnie instrukcji EXECUTE, aby tę instrukcję wykonać. Ewentualnie można użyć instrukcji EXECUTE IMMEDIATE, aby przygotować i wykonać obsługiwaną instrukcję.

Jeśli podczas sesji tranzytowej zostanie użyta komenda COMMIT lub ROLLBACK, spowoduje ona dokończenie bieżącej jednostki pracy, ale nie zakończy sesji tranzytowej.

Uwagi i ograniczenia dotyczące sesji tranzytowych w systemach stowarzyszonych

Korzystając z sesji tranzytowych, należy mieć na uwadze szereg zaleceń i ograniczeń. Poniższe uwagi dotyczą wszystkich źródeł danych:

- Instrukcje przygotowane podczas sesji tranzytywnej muszą być wykonane w tej samej sesji. Próba wykonania instrukcji przygotowanej podczas sesji tranzytywnej poza tą sesją nie powiedzie się i spowoduje błąd SQLSTATE 56098.
- Aplikacja może wielokrotnie użyć instrukcji SET PASSTHRU, jednak tylko ostatnia sesja jest aktywna. Wywołanie nowej instrukcji SET PASSTHRU anuluje poprzednią instrukcję SET PASSTHRU. W ramach jednej sesji tranzytywnej można wymieniać dane tylko z jednym źródłem danych.
- Jeśli w aplikacji używanych jest wiele sesji tranzytowych, należy zawsze użyć instrukcji COMMIT przed uruchomieniem kolejnej sesji. Spowoduje to dokończenie jednostki pracy w bieżącej sesji.
- Znaczniki parametrów nie są obsługiwane w sesjach tranzytowych. Zamiast nich należy używać zmiennych języka bazowego.
- Dopuszczalne jest posługiwanie się semantyką WITH HOLD w stosunku do kursora zdefiniowanego w sesji tranzytywnej. Jednak próba użycia tej semantyki (ze słowem COMMIT) w sytuacji, gdy źródło danych nie obsługuje semantyki WITH HOLD, spowoduje wygenerowanie błędu.
- Zmienne języka bazowego definiowane w instrukcjach SQL podczas sesji tranzytywnej muszą mieć postać: Hn , gdzie H jest pisane wielką literą, a n jest unikalną liczbą całkowitą. n musi przyjmować wartości kolejne, zaczynając od 0.
- W sesjach tranzytowych nie są obsługiwane obiekty LOB.
- W sesjach tranzytowych nie są obsługiwane wywołania procedur zapisanych w bazie.
- W sesjach tranzytowych nie jest obsługiwana instrukcja SELECT INTO.

Pojęcia pokrewne:

- “Sesje tranzytowe” na stronie 10
- “Bezpośrednie kierowanie zapytań do źródeł danych w sesji tranzytywnej” na stronie 213

Zadania pokrewne:

- “Sesje tranzytowe do źródeł danych Oracle” na stronie 214
- “Dostęp do źródeł danych przy użyciu sesji tranzytowych” na stronie 107

Informacje pokrewne:

- “SET PASSTHRU statement” w podręczniku *SQL Reference, Volume 2*

Sesje tranzytowe do źródeł danych Oracle

Przed wprowadzeniem w sesji tranzytywnej instrukcji SQL dotyczących źródła danych Oracle należy wziąć pod uwagę następujące uwarunkowania:

- Gdy zdalny klient używa instrukcji SELECT z procesora wiersza komend (CLP) w trybie tranzytowym i kod klienta pochodzi z pakietu SDK wcześniejszego niż DB2 Universal Database wersja 5, instrukcja SELECT spowoduje błąd SQLCODE -30090 z kodem przyczyny 11. Aby tego uniknąć, zdalni klienci muszą korzystać z pakietu SDK dostępnego w wersji 5 lub nowszej.
- Dowolna instrukcja języka DDL użyta na serwerze Oracle jest wykonywana w czasie analizy i nie podlega semantynom transakcji. Po zakończeniu operacja taka jest automatycznie zatwierdzana przez system Oracle. Jeśli wystąpi wycofanie zmian, instrukcja DDL nie zostanie wycofana.

- Aby instrukcja SELECT dla typów danych źródłowych zwróciła wartości szesnastkowe, należy użyć funkcji RAWTOHEX. Wykonując instrukcję INSERT dla typów danych źródłowych, należy używać reprezentacji szesnastkowej.

Część 5. Informacje dodatkowe

Rozdział 19. Widoki w tabeli katalogu globalnego zawierające informacje o systemie stowarzyszonym

Większość widoków katalogu w stowarzyszonej bazie danych stanowi dokładne odpowiedniki widoków katalogu w dowolnej innej bazie danych DB2 w systemach Linux, UNIX lub Windows. Oprócz nich jednak obecnych jest kilka specjalnych widoków zawierających informacje dotyczące tylko systemu stowarzyszonego, jak widok SYSCAT.WRAPPERS.

Zgodnie z informacjami w podręczniku SQL Reference programu DB2 dla systemów Linux, UNIX i Windows wersja 6 i wersja 7 widoki SYSCAT z programu DB2 wersja 8 są teraz przeznaczone tylko do odczytu. Wszelkie operacje UPDATE i INSERT względem widoku SYSCAT zakończą się niepowodzeniem. Zalecaną metodą modyfikowania katalogu systemowego jest korzystanie z widoków SYSSTAT. Dlatego aplikacje odwołujące się do widoku SYSCAT należy zaktualizować, wprowadzając odwołania do widoku SYSSTAT.

W poniższej tabeli wymieniono widoki SYSCAT zawierające informacje o systemie stowarzyszonym. Są to widoki przeznaczone tylko do odczytu.

Tabela 19. Widoki katalogu używane zwykle w kontekście systemu stowarzyszonego

Widoki katalogu	Opis
SYSCAT.CHECKS	Zawiera informacje o zdefiniowanych ograniczeniach sprawdzających.
SYSCAT.COLCHECKS	Zawiera kolumny, do których odwołuje się ograniczenie sprawdzające.
SYSCAT.COLUMNS	Zawiera informacje o kolumnach zawartych w obiektach źródła danych (tabelach i widokach), dla których utworzono pseudonimy.
SYSCAT.COLOPTIONS	Zawiera informacje na temat wartości opcji kolumn ustawionych dla danego pseudonimu.
SYSCAT.CONSTDEP	Zawiera wykaz zależności dla zdefiniowanych ograniczeń informacyjnych.
SYSCAT.DATATYPES	Zawiera informacje o typach danych DB2, zarówno lokalnych wbudowanych, jak i zdefiniowanych przez użytkownika.
SYSCAT.DBAUTH	Zawiera zestawienie uprawnień w bazie danych przyznanych indywidualnym użytkownikom oraz grupom.
SYSCAT.FUNCMAPOPTIONS	Zawiera informacje o wartościach opcji ustawionych dla odwzorowania funkcji.
SYSCAT.FUNCMAPPINGS	Zawiera odwzorowania funkcji między stowarzyszoną bazą danych a obiektami źródła danych.
SYSCAT.INDEXCOLUSE	Zawiera kolumny należące do indeksu.
SYSCAT.INDEXES	Zawiera specyfikacje indeksów dla obiektów źródła danych.
SYSCAT.KEYCOLUSE	Zawiera kolumny należące do klucza zdefiniowanego przez ograniczenie klucza unikalnego, klucza podstawowego lub klucza obcego.

Tabela 19. Widoki katalogu używane zwykle w kontekście systemu stowarzyszonego (kontynuacja)

Widoki katalogu	Opis
SYSCAT.REFERENCES	Zawiera informacje o zdefiniowanych ograniczeniach referencyjnych.
SYSCAT.ROUTINES	Zawiera lokalne funkcje DB2 zdefiniowane przez użytkownika lub szablony funkcji. Szablony funkcji są używane w odwzorowaniach do funkcji zdefiniowanych w źródle danych.
SYSCAT.REVTYPEMAPPINGS	Ten widok nie jest używany. Wszystkie odwzorowania typów danych są rejestrowane w widoku SYSCAT.TYPEMAPPINGS.
SYSCAT.SERVEROPTIONS	Zawiera informacje o wartościach opcji serwera ustawionych w definicji serwera.
SYSCAT.SERVERS	Zawiera definicje serwerów utworzone dla serwerów źródeł danych.
SYSCAT.TABCONST	Każdy wiersz reprezentuje ograniczenia tabel i pseudonimów typu CHECK, UNIQUE, PRIMARY KEY lub FOREIGN KEY.
SYSCAT.TABLES	Zawiera informacje o wszystkich tworzonych lokalnych tabelach DB2, widokach stowarzyszonych i pseudonimach.
SYSCAT.TYPEMAPPINGS	Zawiera proste odwzorowania typów danych i zwrotne odwzorowania typów danych. Są to odwzorowania typów danych w źródle danych na lokalne typy DB2. Odwzorowania te są używane podczas tworzenia pseudonimu dla obiektu źródła danych.
SYSCAT.USEROPTIONS	Zawiera dane dotyczące autoryzacji użytkowników, konfigurowane przy tworzeniu odwzorowań użytkowników między serwerem stowarzyszonej bazy danych a serwerem źródła danych.
SYSCAT.VIEWS	Zawiera informacje o tworzonych lokalnych widokach stowarzyszonych.
SYSCAT.WRAPOPTIONS	Zawiera informacje na temat wartości opcji ustawionych dla opakowania.
SYSCAT.WRAPPERS	Zawiera nazwę opakowania i plik biblioteki dla każdego źródła danych, dla którego utworzono opakowanie.

W poniższej tabeli wymieniono widoki SYSSTAT zawierające informacje o systemie stowarzyszonym. Są to widoki umożliwiające zarówno odczyt, jak i zapis; zawierają dane statystyczne możliwe do modyfikacji.

Tabela 20. Widoki stowarzyszone katalogu globalnego umożliwiające modyfikację

Widoki katalogu	Opis
SYSSTAT.COLUMNS	Zawiera informacje statystyczne o kolumnach zawartych w obiektach źródła danych (tabelach i widokach), dla których utworzono pseudonimy. Statystyki nie są generowane w przypadku kolumn dziedziczonych tabel określonego typu.

Tabela 20. Widoki stowarzyszone katalogu globalnego umożliwiające modyfikację (kontynuacja)

Widoki katalogu	Opis
SYSSTAT.INDEXES	Zawiera informacje statystyczne na temat wszystkich specyfikacji indeksów dla obiektów źródła danych.
SYSSTAT.ROUTINES	Zawiera informacje statystyczne o każdej z funkcji zdefiniowanych przez użytkownika. Nie obejmuje funkcji wbudowanych. Statystyki nie są generowane w przypadku kolumn dziedziczonych tabel określonego typu.
SYSSTAT.TABLES	Zawiera informacje na temat wszystkich tabel bazowych. Widok nie obejmuje informacji o widokach, synonimach i aliasach. W przypadku tabel określonego typu widok obejmuje tylko tabelę główną hierarchii tabel. Statystyki nie są generowane w przypadku kolumn dziedziczonych tabel określonego typu.

Rozdział 20. Opcje opakowań w systemie stowarzyszonym

Opcje opakowań służą do konfigurowania opakowań lub do definiowania sposobu, w jaki serwer stowarzyszony używa opakowania. Opcje opakowania określa się podczas tworzenia lub modyfikowania opakowania.

Wszystkie relacyjne i nierelacyjne źródła danych używają opcji opakowania DB2_FENCED. Źródła danych ODBC używają opcji opakowania MODULE. Źródła danych Entrez używają opcji opakowania EMAIL.

Tabela 21. Opcje opakowań wraz z ustawieniami

Opcja	Poprawne ustawienia	Ustawienie domyślne
DB2_FENCED	Określa, czy opakowanie jest uruchamiane w trybie chronionym czy zaufanym.	Opakowania relacyjne: N.
	Y Opakowanie jest uruchamiane w trybie chronionym.	Opakowania nierelacyjne firmy IBM: N.
	N Opakowanie jest uruchamiane w trybie zaufanym.	Opakowania nierelacyjne innych firm: Y.
EMAIL	Określa adres e-mail używany podczas rejestrowania opakowania Entrez. Ten adres jest dołączany do wszystkich zapytań i umożliwia systemowi NCBI wysyłanie powiadomień w razie wystąpienia problemów, na przykład przeciążenia serwerów NCBI zbyt wieloma zapytaniami. Ta opcja jest wymagana.	
MODULE	Określa pełną ścieżkę do biblioteki zawierającej implementację Menedżera sterowników ODBC lub implementację SQL/CLI. Wymagane dla opakowań ODBC na serwerach stowarzyszonych systemu UNIX.	W systemach Windows wartość domyślna to odbc32.dll.

Pojęcia pokrewne:

- “Paralelizm i zapytania odwołujące się do pseudonimów” na stronie 157

Zadania pokrewne:

- “Trusted and fenced mode process environments” w podręczniku *IBM DB2 Information Integrator Wrapper Developer’s Guide*
- “Modyfikowanie opakowania” na stronie 27
- “Registering wrappers for a data source” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Rozdział 21. Opcje serwera w systemie stowarzyszonym

Opcje serwera służą do opisu serwera źródła danych. Opcje serwera określają informacje o integralności danych, ich położeniu, zabezpieczeniach i wydajności. Niektóre z opcji są dostępne dla wszystkich źródeł danych, podczas gdy inne mogą być używane tylko w kontekście niektórych źródeł.

Wspólne opcje serwerów stowarzyszonych dla źródeł relacyjnych, to:

- Opcje zgodności: COLLATING_SEQUENCE, IGNORE_UDT
- Opcje integralności danych: IUD_APP_SVPT_ENFORCE
- Opcje daty i godziny: DATEFORMAT, TIMEFORMAT, TIMESTAMPFORMAT
- Opcje położenia: CONNECTSTRING, DBNAME, IFILE
- Opcje bezpieczeństwa: FOLD_ID, FOLD_PW, INFORMIX_LOCK_MODE
- Opcje wydajności: COMM_RATE, CPU_RATIO, DB2_MAXIMAL_PUSHDOWN, IO_RATIO, LOGIN_TIMEOUT, PACKET_SIZE, PLAN_HINTS, PUSHDOWN, TIMEOUT, VARCHAR_NO_TRAILING_BLANKS

Następująca tabela zawiera wykaz opcji definiujących serwery, mających zastosowanie do wszystkich źródeł relacyjnych.

Tabela 22. Opcje serwera dla relacyjnych źródeł danych

Źródło danych	CODEPAGE	COLLATING_SEQUENCE	COMM_RATE	CONNECTSTRING	CPU_RATIO	DATEFORMAT	DB2_MAXIMAL_PUSHDOWN	DBNAME	FOLD_ID	FOLD_PW	IFILE	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 UDB for iSeries		X	X		X		X	X	X	X			X	X				X		X				X
DB2 UDB for z/OS and OS/390		X	X		X		X	X	X	X			X	X				X		X				X
DB2 for VM and VSE		X	X		X		X	X	X	X			X	X				X		X				X
DB2 UDB dla systemów Linux, UNIX i 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				
Microsoft SQL Server	X	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	X	X

Tabela 22. Opcje serwera dla relacyjnych źródeł danych (kontynuacja)

Źródło danych	CODEPAGE	COLLATING_SEQUENCE	COMM_RATE	CONNECTSTRING	CPU_RATIO	DATEFORMAT	DB2_MAXIMAL_PUSHDOWN	DBNAME	FOLD_ID	FOLD_PW	IFILE	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
OLE DB		X		X																				
Oracle		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			
Teradata		X	X		X		X						X	X		X				X				

Poniższa tabela zawiera listę opcji definicji serwera dla wszystkich nierelacyjnych źródeł danych z wyjątkiem WebSphere Business Integration. Opcje definicji serwera dla źródeł danych WebSphere Business Integration zawiera Tabela 24 na stronie 227.

Tabela 23. Opcje serwera dla nierelacyjnych źródeł danych.

Źródło danych	CASE_SENSITIVE	CONTENT_DIR	DAEMON_PORT	ES_HOST	ES_PORT	ES_TRACING	ES_TRACELEVEL	ES_TRACEFILENAME	HMMPFAM_OPTIONS	HMMSEARCH_OPTIONS	MAX_ROWS	NODE	OS_TYPE	PORT	PROCESSORS	PROXU_AUTHID	PROXY_PASSWORD	PROXY_SERVER_NAME	PROXY_SERVER_PORT	PROXY_TYPE	RDBMS_TYPE	SOCKET_TIMEOUT	TIMEOUT	TRANSACTIONS	USE_CLOB_SEQUENCE
BioRS	X											X		X									X		
BLAST			X									X													X
Documentum		X										X	X								X			X	
Entrez											X					X	X	X	X	X		X			
Excel																									
Extended Search				X	X	X	X	X																	
HMMER			X						X	X		X			X										X
Pliki o strukturze tabeli																									
Usługi WWW																									
XML																X	X	X	X	X		X			

Poniższa tabela zawiera listę opcji definicji serwera dla źródeł danych WebSphere Business Integration.

Tabela 24. Opcje definicji serwera dla źródeł danych WebSphere Business Integration

Źródło danych	APP_TYPE	FAULT_QUEUE	MQ_CONN_NAME	MQ_MANAGER	MQ_RESPONSE_TIMEOUT	MQ_SVRCONN_CHANNELNAME	REQUEST_QUEUE	RESPONSE_QUEUE
WebSphere Business Integration	X	X	X	X	X	X	X	X

W poniższej tabeli zestawiono opisy wszystkich opcji serwera wraz z poprawnymi i domyślnymi ustawieniami.

Tabela 25. Opcje serwera z ustawieniami

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
APP_TYPE	Typ zdalnej aplikacji. Poprawne wartości to 'PSOFT', 'SAP' i 'SIEBEL'. Ta opcja jest wymagana.	Brak.

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
CASE_SENSITIVE	<p>Określa, czy serwer BioRS rozróżnia wielkość liter w nazwach. Poprawne wartości to Y i N.</p> <p>'Y' Serwer BioRS rozróżnia wielkość liter w nazwach.</p> <p>'N' Serwer BioRS nie rozróżnia wielkości liter w nazwach.</p> <p>W produkcie BioRS rozróżnianie wielkości liter w danych przechowywanych na serwerze BioRS zależy od ustawienia parametru konfiguracyjnego. Opcja CASE_SENSITIVE to odpowiednik tego parametru w programie DB2 Information Integrator. Wymagane jest zsynchronizowanie ustawień konfiguracyjnych dotyczących rozróżniania wielkości liter w systemie BioRS i programie DB2 Information Integrator. Jeśli ustawienia konfiguracyjne dotyczące rozróżniania wielkości liter w serwerze BioRS i programie DB2 Information Integrator nie będą zsynchronizowane, podczas próby dostępu do danych na serwerze BioRS za pośrednictwem programu DB2 Information Integrator wystąpi błąd.</p> <p>Po utworzeniu nowego serwera BioRS w programie DB2 Information Integrator nie można zmienić ani usunąć opcji CASE_SENSITIVE. Gdy zajdzie potrzeba zmiany opcji CASE_SENSITIVE, konieczne będzie usunięcie serwera i ponowne utworzenie go. W przypadku usunięcia serwera BioRS konieczne jest także ponowne utworzenie wszystkich pseudonimów odpowiadających danemu serwerowi BioRS. Program DB2 Information Integrator automatycznie usunie wszystkie pseudonimy odpowiadające usuniętemu serwerowi.</p>	T
CODEPAGE	<p>Określa identyfikator strony kodowej programu DB2, odpowiadający zestawowi znaków kodowych określonego w konfiguracji klienta źródła danych. Określenie strony kodowej klienta jest konieczne, gdy jest ona inna niż strona kodowa stowarzyszonej bazy danych.</p> <p>W przypadku źródeł danych obsługujących Unicode, opcji CODEPAGE można nadać wartość identyfikatora strony kodowej programu DB2 odpowiadającego kodowaniu Unicode obsługiwaneemu przez klienta źródła danych.</p>	<p>W systemach UNIX lub Windows ze stowarzyszoną bazą danych, która nie obsługuje Unicode: strona kodowa stowarzyszonej bazy danych.</p> <p>W systemach UNIX ze stowarzyszoną bazą danych, która obsługuje Unicode: 1208</p> <p>W systemach Windows ze stowarzyszoną bazą danych, która obsługuje Unicode: 1202</p>

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
COLLATING_SEQUENCE	<p>Określa, czy źródło danych używa tej samej kolejności zestawiania co stowarzyszona baza danych, odpowiednio do zestawu kodowego NLS i informacji o kraju lub regionie.</p> <p>'Y' W źródle danych obowiązuje taka sama kolejność zestawiania, jak w stowarzyszonej bazie danych DB2.</p> <p>'N' W źródle danych obowiązuje inna kolejność zestawiania niż w stowarzyszonej bazie danych DB2.</p> <p>'I' W źródle danych obowiązuje inna kolejność zestawiania niż w stowarzyszonej bazie danych DB2, a ponadto w kolejności zestawiania w źródle danych nie jest brana pod uwagę wielkość liter (na przykład ciągi 'STEWART' i 'StewART' są rozpoznawane jako jednakowe).</p>	'N'
COMM_RATE	<p>Określa szybkość komunikacji między serwerem stowarzyszonym a serwerem źródła danych. Parametr wyrażony w megabajtach na sekundę.</p> <p>Dopuszczalne są wartości większe niż 0 i mniejsze niż 1×10^{23}. Wartości mogą być wyrażane w dowolnej notacji poprawnej dla typu REAL.</p>	'2'
CONTENT_DIR	<p>Określa nazwę lokalnie dostępnego katalogu głównego służącego do przechowywania plików danych pobranych przez pseudokolumny GET_FILE, GET_FILE_DEL, GET_RENDITION i GET_RENDITION_DEL. Prawa zapisu do tego katalogu muszą mieć wszyscy użytkownicy, którzy mogą korzystać z pseudokolumn.</p>	<p>W systemach UNIX: '/tmp'</p> <p>W systemach Windows: 'C:\tmp'</p>
CONNECTSTRING	<p>Określa właściwości wymagane do zainicjowania połączenia z dostawcą OLE DB.</p>	Brak.
CPU_RATIO	<p>Określa, o ile szybciej lub wolniej działa procesor źródła danych w stosunku do procesora serwera stowarzyszonego.</p> <p>Dopuszczalne są wartości większe niż 0 i mniejsze niż 1×10^{23}. Wartości mogą być wyrażane w dowolnej notacji poprawnej dla typu REAL.</p> <p>Ustawienie 1 oznacza, że szybkości procesora serwera stowarzyszonego DB2 i procesora źródła danych są jednakowe, co wyraża się stosunkiem 1:1. Ustawienie .5 oznacza, że procesor serwera stowarzyszonego DB2 jest o 50% wolniejszy niż procesor źródła danych. Ustawienie 2 oznacza, że procesor serwera stowarzyszonego DB2 jest dwa razy szybszy niż procesor źródła danych.</p>	'1.0'
DATEFORMAT	<p>Format daty używany w źródle danych. Format należy zdefiniować za pomocą członów 'DD', 'MM' i 'YY' lub 'YYYY', reprezentujących odpowiednie części zapisu daty. Należy także określić ogranicznik w rodzaju spacji lub przecinka. Na przykład, aby zdefiniować format daty '2003-01-01', należy użyć zapisu 'YYYY-MM-DD'. W tym polu jest dopuszczalna wartość pusta.</p>	Brak.

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
DAEMON_PORT	Określa numer portu, na którym demon będzie nasłuchiwał żądań zadań BLAST lub HMMER. Numer portu musi być taki sam, jak numer określony w opcji DAEMON_PORT w pliku konfiguracyjnym demona.	BLAST: '4007'; HMMER: '4098'
DB2_MAXIMAL_PUSHDOWN	Określa podstawowe kryteria stosowane przez optymalizator zapytań przy doborze planu dostępu. Optymalizator zapytań może dobrać plan dostępu na podstawie kosztu lub postawionego przez użytkownika warunku, aby jak największa część obciążenia przetwarzaniem przypadła na zdalne źródło danych. 'Y' Optymalizator zapytań wybiera taki plan dostępu, który przewiduje przekazanie największej liczby operacji do źródła danych. W przypadku gdy kilka planów dostępu pozwala osiągnąć jednakowy poziom przekazywania, optymalizator zapytań wybiera plan o najniższym koszcie. Jeśli przynajmniej część zapytania może być przetworzona przy użyciu zmaterializowanej tabeli zapytania na serwerze stowarzyszonym, może być wybrany plan uwzględniający taką możliwość. Stowarzyszona baza danych nie przekazuje niżej zapytań, których wynikiem jest iloczyn kartezjański. 'N' Optymalizator zapytań doбира plan dostępu na podstawie kosztu.	'N'
DBNAME	Nazwa bazy danych źródła danych, do której ma się odwoływać serwer stowarzyszony. Dla bazy danych DB2 wartość ta odpowiada konkretnej bazie danych używanej podczas początkowego połączenia ze zdalną bazą danych DB2. Ta konkretna baza danych to alias zdalnej bazy danych DB2 wpisany do katalogu na serwerze stowarzyszonym przy użyciu komendy CATALOG DATABASE lub Asysty podczas konfigurowania DB2. Nie dotyczy źródeł danych Oracle, ponieważ instancje Oracle zawierają tylko jedną bazę danych.	Brak.
ES_HOST	Określa pełną nazwę hosta lub adres IP serwera Extended Search, którego dotyczy wyszukiwanie. Ta opcja jest wymagana.	Brak.
ES_PORT	Określa numer portu, pod którym serwer Extended Search nasłuchuje żądań. Ta opcja nie jest wymagana.	'6001'
ES_TRACING	Określa, czy ma być włączone śledzenie komunikatów o błędach i komunikatów informacyjnych generowanych przez zdalny serwer Extended Search. Poprawne wartości to: 'OFF' Nie będą protokołowane żadne komunikaty. 'ON' Komunikaty śledzenia będą protokołowane. Ta opcja nie jest wymagana.	'OFF'

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
ES_TRACELEVEL	<p>Gdy śledzenie jest włączone, ta opcja określa typy komunikatów, które będą zapisywane w pliku protokołu. Poniższe poziomy śledzenia można włączać i wyłączać niezależnie:</p> <p>'C' Komunikaty o błędach krytycznych.</p> <p>'N' Komunikaty o błędach innych niż krytyczne.</p> <p>'W' Komunikaty ostrzegawcze.</p> <p>'I' Komunikaty informacyjne.</p> <p>Na przykład:</p> <p>ES_TRACELEVEL 'W' ES_TRACELEVEL 'CN'</p> <p>Ta opcja nie jest wymagana.</p>	'C'
ES_TRACEFILENAME	<p>Gdy śledzenie jest włączone, ta opcja określa nazwę katalogu i pliku, w którym komunikaty będą zapisywane. Ta opcja nie jest wymagana.</p>	<p>Dla systemów operacyjnych UNIX: \$INSTHOME/sql/lib/log/ESWrapper.log.</p> <p>Dla systemów operacyjnych Windows: %DB2TEMPDIR%\ESWrapper.log.</p>
FAULT_QUEUE	<p>Nazwa kolejki błędów, która dostarcza komunikaty o błędach z adaptera do opakowania. Nazwa musi być zgodna ze specyfikacjami programu WebSphere MQ dla nazw kolejek. Opcja ta jest wymagana.</p>	Brak.
FOLD_ID (Patrz przypisy 1 i 4 na końcu tabeli).	<p>Dotyczy identyfikatorów użytkownika wysyłanych przez serwer stowarzyszony na serwer źródła danych w celu uwierzytelnienia. Poprawne wartości to:</p> <p>'U' Przed wysłaniem identyfikatora do źródła danych serwer stowarzyszony przekształca wszystkie jego znaki na wielkie litery. Jest to poprawne ustawienie w przypadku źródeł danych z rodziny DB2 i Oracle (patrz przypis 2 na końcu tabeli).</p> <p>'N' Serwer stowarzyszony nie wykonuje żadnych operacji na identyfikatorze przed wysłaniem go do źródła danych (patrz przypis 2 na końcu tabeli).</p> <p>'L' Przed wysłaniem identyfikatora do źródła danych serwer stowarzyszony przekształca wszystkie jego znaki na małe litery.</p> <p>Gdy nie jest wybrane żadne z tych ustawień, serwer stowarzyszony próbuje wysłać identyfikator w postaci wielkich liter. Jeśli uwierzytelnienie się nie powiedzie, serwer ponawia wysłanie identyfikatora z użyciem małych liter.</p>	Brak.

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
FOLD_PW (Patrz przypisy 1, 3 i 4 na końcu tabeli).	<p>Dotyczy haseł wysyłanych przez serwer stowarzyszony do źródła danych w celu dokonania uwierzytelnienia. Poprawne wartości to:</p> <p>'U' Przed wysłaniem hasła do źródła danych serwer stowarzyszony przekształca wszystkie jego znaki na wielkie litery. Jest to ustawienie poprawne w przypadku źródeł danych z rodziny DB2 i Oracle.</p> <p>'N' Serwer stowarzyszony nie wykonuje żadnych operacji na hasle przed wysłaniem go do źródła danych.</p> <p>'L' Przed wysłaniem hasła do źródła danych serwer stowarzyszony przekształca wszystkie jego znaki na małe litery.</p> <p>Gdy nie jest wybrane żadne z tych ustawień, serwer stowarzyszony próbuje wysłać hasło w postaci wielkich liter. Jeśli uwierzytelnienie się nie powiedzie, serwer ponawia wysłanie hasła z użyciem małych liter.</p>	Brak.
HMPFAM_OPTIONS	<p>Określa opcje programu hmmpfam, takie jak --null2, --pvm i --xnu, dla których nie ma odpowiadających im nazw kolumn w tabeli pomocniczej, odwzorowującej opcje na nazwy kolumn.</p> <p>Na przykład: HMPFAM_OPTIONS '--xnu --pvm'</p> <p>W tym przykładzie demon uruchomi program HMMPFAM z opcjami z klauzuli WHERE zapytania oraz dodatkowymi opcjami --xnu --pvm.</p>	
HMMSEARCH_OPTIONS	Umożliwia użytkownikom określenie dodatkowych opcji wiersza komend dla komendy hmmsearch. Opcja ta jest poprawna tylko dla typu SEARCH. Więcej informacji na ten temat zawiera podręcznik użytkownika programu HMMER.	Brak.
IFILE	Określa ścieżkę i nazwę pliku interfejsów programu Sybase Open Client. Na serwerach stowarzyszonych z systemem Windows NT jest to domyślnie ścieżka %DB2PATH%\interfaces. Na serwerach stowarzyszonych z systemem UNIX domyślna wartość ścieżki i nazwy to \$DB2INSTANCE/sqllib/interfaces.	Brak.

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
INFORMIX_LOCK_MODE	<p>Określa tryb blokady ustawianej dla źródeł danych Informix. Opakowanie Informix wydaje komendę 'SET LOCK MODE' natychmiast po nawiązaniu połączenia z źródłem danych Informix. Poprawne wartości to:</p> <p>'W' Wybrany jest tryb blokady Informix WAIT. Gdy opakowanie próbuje uzyskać dostęp do zablokowanej tabeli lub wiersza, serwer Informix czeka na zwolnienie blokady.</p> <p>'N' Wybrany jest tryb blokady Informix NOWAIT. Gdy opakowanie próbuje uzyskać dostęp do zablokowanej tabeli lub wiersza, serwer Informix zwraca błąd.</p> <p>'n' Wybrany jest tryb blokady Informix WAIT z oczekiwaniem przez <i>n</i> sekund. Gdy opakowanie próbuje uzyskać dostęp do zablokowanej tabeli lub wiersza, a zwolnienie blokady nie nastąpi w ciągu odpowiedniej liczby sekund, serwer Informix zwraca błąd.</p>	'W'
IO_RATIO	<p>Określa, o ile szybciej lub wolniej działa podsystem we/wy źródła danych w stosunku do podsystemu we/wy serwera stowarzyszonego.</p> <p>Dopuszczalne są wartości większe niż 0 i mniejsze niż 1×10^{23}. Wartości mogą być wyrażane w dowolnej notacji poprawnej dla typu REAL.</p> <p>Ustawienie 1 oznacza, że szybkości podsystemów we/wy serwera stowarzyszonego DB2 i źródła danych są jednakowe, co wyraża się stosunkiem 1:1. Ustawienie .5 oznacza, że podsystem we/wy serwera stowarzyszonego DB2 jest o 50% wolniejszy niż podsystem we/wy źródła danych. Ustawienie 2 oznacza, że podsystem we/wy serwera stowarzyszonego DB2 jest dwa razy szybszy niż podsystem we/wy źródła danych.</p>	'1.0'
IUD_APP_SVPT_ENFORCE	<p>Określa, czy system stowarzyszony DB2 powinien wymuszać wykrywanie lub tworzenie instrukcji punktu zapisu aplikacji. Przy korzystaniu z instrukcji SET SERVER OPTION ta opcja serwera nie będzie miała żadnego wpływu na statyczne instrukcje SQL.</p> <p>'Y' Serwer stowarzyszony wycofuje transakcje wstawienia, aktualizacji i usunięcia, jeśli wystąpi w nich błąd, a źródło danych nie wymusi instrukcji punktu zapisu aplikacji. Zostanie zwrócony kod błędu SQL SQL1476N.</p> <p>'N' Serwer stowarzyszony nie będzie wycofywał transakcji w razie napotkania błędu. Za obsługę błędów odpowiada aplikacja.</p>	'Y'
LOGIN_TIMEOUT	<p>Określa liczbę sekund, jaką serwer stowarzyszony DB2 może oczekiwać na odpowiedź programu Sybase Open Client na wysłane żądanie logowania. Domyślnie jest to okres taki sam jak dla ustawienia TIMEOUT.</p>	'0'

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
MAX_ROWS	<p>Określa liczbę wierszy zwracanych przez serwer stowarzyszony dla zapytania używającego opakowania Entrez.</p> <p>Można podać tylko liczbę dodatnią lub zero. Nadanie tej opcji wartości zero umożliwi zapytaniom pobieranie nieograniczonej liczby wierszy z serwisu WWW NCBI. Jednak nadanie opcji serwera MAX_ROWS wartości zero lub wartości bardzo dużej może niekorzystnie wpłynąć na wydajność zapytań.</p> <p>Opcja serwera MAX_ROWS nie jest wymagana.</p>	<p>Systemy operacyjne Microsoft Windows: 2000 wierszy.</p> <p>Systemy operacyjne typu UNIX: 5000 wierszy.</p>
MQ_CONN_NAME	Nazwa hosta lub adres sieciowy komputera, na którym działa serwer Websphere MQ. Przykładem nazwy połączenia może być: 9.30.76.151(1420), gdzie 1420 jest numerem portu. Jeśli nie zostanie określony numer portu, używany będzie port o domyślnym numerze 1414. Ta opcja nie jest wymagana. Jeśli zostanie pominięta, do wyboru definicji kanału zostanie użyta zmienna środowiskowa MQSERVER (o ile została określona w pliku db2dj.ini). Jeśli nie określono zmiennej MQSERVER, zostanie użyta tabela kanałów klienta.	Opakowanie używa zmiennej środowiskowej MQSERVER, o ile została ona określona w pliku db2dj.ini, do wyboru definicji kanału. Jeśli nie określono zmiennej MQSERVER, opakowanie użyje tabeli kanałów klienta.
MQ_MANAGER	Nazwa menedżera WebSphere MQ. Dowlolna poprawna nazwa menedżera WebSphere MQ. Ta opcja jest wymagana.	Brak.
MQ_RESPONSE_TIMEOUT	Czas oczekiwania przez opakowanie na komunikat odpowiedzi z kolejki odpowiedzi. Wartość podawana jest w milisekundach. Opcji można nadać wartość specjalną -1, aby określić brak limitu czasu. Ta opcja nie jest wymagana.	10000
MQ_SVRCONN_CHANNELNAME	Nazwa kanału połączenia z serwerem w menedżerze Websphere MQ Manager, z którym opakowanie powinno próbować się połączyć. Ten parametr można podać tylko wtedy, gdy podano również opcję serwera MQ_CONN_NAME. W przypadku pominięcia tej opcji zostanie użyty domyślny kanał połączenia z serwerem, SYSTEM.DEF.SVRCONN.	SYSTEM.DEF.SVRCONN
NODE	<p>Relacyjne źródła danych: nazwa, pod którą źródło danych jest zdefiniowane jako instancja w swoim systemie RDBMS.</p> <p>Documentum: określa rzeczywistą nazwę bazy danych Documentum Docbase. Ta opcja jest wymagana.</p> <p>BLAST: określa nazwę hosta systemu, w którym działa proces demona BLAST. Ta opcja jest wymagana.</p> <p>HMMER: określa nazwę hosta serwera, na którym działa proces demona HMMER. Ta opcja jest wymagana.</p> <p>BioRS: określa nazwę hosta systemu, w którym dostępne jest narzędzie zapytań BioRS. Ta opcja nie jest wymagana.</p>	BioRS: <i>localhost</i>
OS_TYPE	Określa system operacyjny serwera Docbase. Poprawne wartości to AIX, SOLARIS i WINDOWS. Ta opcja jest wymagana.	Brak.

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
PACKET_SIZE	Określa w bajtach wielkość pakietów pliku interfejsów Sybase. Jeśli źródło danych nie obsługuje pakietów o podanej wielkości, połączenie zakończy się niepowodzeniem. Powiększenie pakietów może wydatnie poprawić wydajność w przypadku dużych rekordów (na przykład przy wstawianiu wierszy do dużych tabel). Wielkość jest określana w postaci liczby bajtów.	
PASSWORD	Określa, czy do źródła danych wysyłane są hasła. 'Y' Hasła są wysyłane do źródła danych i sprawdzana jest ich poprawność. 'N' Hasła nie są wysyłane do źródła danych, a ich poprawność nie jest sprawdzana.	'Y'
PLAN_HINTS	Określa, czy mają być uwzględniane <i>wskazówki dotyczące planu</i> . Wskazówki dotyczące planu są to fragmenty instrukcji, które dostarczają dodatkowych informacji optymalizatorom źródeł danych. Informacje takie mogą poprawić wydajność niektórych rodzajów zapytań. Na podstawie wskazówek dotyczących planu optymalizator może łatwiej zdecydować, czy należy korzystać z indeksu, który to powinien być indeks, oraz jaką kolejność łączenia tabel należy zastosować. 'Y' Dla danego źródła danych obsługa wskazówek dotyczących planu będzie włączona, jeśli tylko źródło zezwala na ich stosowanie. 'N' Wskazówki dotyczące planu nie są włączone dla tego źródła danych. Ta opcja jest dostępna tylko dla źródeł danych Oracle i Sybase.	'N'
PORT	Określa numer portu używanego przez opakowanie do połączenia z serwerem BioRS. Ta opcja nie jest wymagana.	'5014'
PROCESSORS	Określa liczbę procesorów używanych przez program HMMER. Ta opcja jest równoważna opcji <code>--cpu</code> komendy <code>hmmpfam</code> .	Brak.
PROXY_AUTHID	Określa nazwę użytkownika używaną, gdy opcja PROXY_TYPE ma wartość 'SOCKS5'. To pole jest opcjonalne, gdy opcja PROXY_TYPE ma wartość 'SOCKS5'. W celu uzyskania odpowiedniej nazwy użytkownika należy skontaktować się z administratorem sieci. Ta opcja jest niepoprawna, gdy opcja PROXY_TYPE ma wartość inną niż 'SOCKS5'.	Brak.
PROXY_PASSWORD	Określa hasło używane, gdy opcja PROXY_TYPE ma wartość 'SOCKS5'. To pole jest opcjonalne, gdy opcja PROXY_TYPE ma wartość 'SOCKS5'. W celu uzyskania odpowiedniego hasła należy skontaktować się z administratorem sieci. Ta opcja jest niepoprawna, gdy opcja PROXY_TYPE ma wartość inną niż 'SOCKS5'.	Brak.

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
PROXY_SERVER_NAME	Określa nazwę lub adres IP serwera proxy. To pole jest wymagane, gdy opcja PROXY_TYPE ma wartość 'HTTP', 'SOCKS4' lub 'SOCKS5'. W celu uzyskania odpowiedniej nazwy serwera proxy lub jego adresu IP należy skontaktować się z administratorem sieci.	Brak.
PROXY_SERVER_PORT	Określa numer portu serwera proxy. To pole jest wymagane, gdy opcja PROXY_TYPE ma wartość 'HTTP', 'SOCKS4' lub 'SOCKS5'. W celu uzyskania numeru portu serwera proxy należy skontaktować się z administratorem sieci.	Brak.
PROXY_TYPE	Określa typ serwera proxy używany w celu uzyskania dostępu do Internetu zza firewalla. Poprawne wartości to 'NONE', 'HTTP', 'SOCKS4' lub 'SOCKS5'. Wartość domyślna to 'NONE'. W celu uzyskania informacji o typie serwera proxy należy skontaktować się z administratorem sieci.	'NONE'
PUSHDOWN	'Y' Program DB2 uwzględni możliwość przekazywania operacji wartościowania do źródła danych. 'N' Program DB2 UDB będzie wysyłać do źródła danych wyłącznie instrukcje SQL SELECT z nazwami kolumn. Predykaty (np. WHERE=), funkcje kolumnowe i funkcje skalarne (np. MAX i MIN), deklaracje sortowania (np. ORDER BY lub GROUP BY) oraz łączenia nie będą uwzględniane w kodzie SQL wysłanym do źródła danych.	'Y'
RDBMS_TYPE	Określa system RDBMS używany przez serwer Docbase. Poprawne wartości to DB2, INFORMIX, ORACLE, SQLSERVER lub SYBASE. Ta opcja jest wymagana.	Brak.
RESPONSE_QUEUE	Nazwa kolejki odpowiedzi, która dostarcza wyniki zapytania z adaptera do opakowania. Nazwa musi być zgodna ze specyfikacjami programu WebSphere MQ dla nazw kolejek. Ta opcja jest wymagana.	Brak.
REQUEST_QUEUE	Nazwa kolejki żądań, która dostarcza żądania zapytania z opakowania do adaptera. Nazwa musi być zgodna ze specyfikacjami programu WebSphere MQ dla nazw kolejek. Ta opcja jest wymagana.	Brak.
SOCKET_TIMEOUT	Określa - w minutach - maksymalny czas oczekiwania stowarzyszonego serwera DB2 na wyniki z serwera proxy. Poprawną wartością jest dowolna liczba nieujemna. Wartością domyślną jest zero ('0'). Wartość zero oznacza nieograniczony czas oczekiwania.	0
TIMEFORMAT	Format godziny używany w źródle danych. Format należy określić przy użyciu członów 'hh12', 'hh24', 'mm', 'ss', 'AM' lub 'A.M'. Na przykład, aby zdefiniować format godziny '16:00:00', należy użyć zapisu 'hh24:mm:ss'. Aby zdefiniować format godziny '8:00:00 AM', należy użyć zapisu 'hh12:mm:ss AM'. W tym polu jest dopuszczalna wartość pusta.	Brak.

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
TIMESTAMPFORMAT	Format datownika używany w źródle danych. Składa się ze specyfikacji formatu daty i godziny, po których następuje wartość 'n' dla dziesiątych części sekundy, 'nn' dla setnych części sekundy, 'nnn' dla milisekund i tak dalej, aż do 'nnnnnn' dla mikrosekund. Na przykład, aby zdefiniować format datownika '2003-01-01-24:00:00.000000', należy użyć zapisu 'YYYY-MM-DD-hh24:mm:ss.nnnnnn'. W tym polu jest dopuszczalna wartość pusta.	Brak.
TIMEOUT	Sybase: określa w sekundach czas oczekiwania serwera stowarzyszonego DB2 na odpowiedź programu Sybase Open Client na dowolną instrukcję SQL. Wartość wyrażona w <i>sekundach</i> jest dodatnią liczbą całkowitą w zakresie liczb całkowitych programu DB2 Universal Database. Podana wartość limitu czasu zależy od używanego opakowania. Domyślnym ustawieniem opcji TIMEOUT dla opakowań Sybase jest 0, co oznacza, że program DB2 UDB może oczekiwać na odpowiedź dowolnie długo. BioRS: określa - w minutach - czas oczekiwania opakowania BioRS na odpowiedź z serwera BioRS. Wartość domyślna wynosi 10. Ta opcja nie jest wymagana.	'0'; BioRS: '10'
TRANSACTIONS	Określa tryb transakcji serwera. Poprawne wartości to: 'NONE' Transakcje są wyłączone. 'QUERY' Transakcje są włączone tylko dla metod Dctm_Query. 'ALL' Transakcje są włączone dla metody Dctm_Query. W tej wersji wartość ALL pełni taką samą funkcję, jak wartość QUERY.	'QUERY'
USE_CLOB_SEQUENCE	Ta opcja określa typ danych używanych przez serwer stowarzyszony dla kolumny BlastSeq lub HmmQSeq. Poprawne wartości to 'Y' i 'N'. W celu przesłonięcia domyślnego typu danych dla kolumn BlastSeq lub HmmQSeq można użyć instrukcji CREATE NICKNAME lub ALTER NICKNAME.	'Y'

Tabela 25. Opcje serwera z ustawieniami (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
VARCHAR_NO_ TRAILING_BLANKS	<p>Ta opcja ma zastosowanie do źródeł danych typu znakowego o zmiennej długości, w których długość nie jest dopełniana podczas porównywania odstępami końcowymi.</p> <p>Niektóre źródła danych, jak Oracle, nie mają semantyki zapewniającej porównywanie danych znakowych dopełnianych spacjami z uzyskaniem takich samych wyników, jak porównywanie w programie DB2 dla systemów Linux, UNIX i Windows. Opcję tę należy ustawić, jeśli ma ona się odnosić do wszystkich kolumn VARCHAR i VARCHAR2 w obiektach źródła danych, do których wyznaczony serwer będzie się odwoływał. Dotyczy to także widoków.</p> <p>Y W takich kolumnach typu VARCHAR nie ma końcowych znaków odstepu lub źródło danych ma podobną semantykę porównywania danych dopełnianych odstępami, jak serwer stowarzyszony.</p> <p>Serwer stowarzyszony przekazuje operacje porównywania danych znakowych do źródła danych w celu ich przetworzenia.</p> <p>N W takich kolumnach typu VARCHAR występują końcowe znaki odstepu, a źródło danych ma semantykę porównywania danych dopełnianych odstępami inne niż serwer stowarzyszony.</p> <p>Jeśli kompensacja do równoważnych semantyk jest niemożliwa, serwer stowarzyszony przetwarza operacje porównywania znaków. Na przykład przebudowuje predykat.</p>	N dla niezgodnych źródeł danych

Przypisy do tabeli:

- To pole jest stosowane bez względu na wartość użytą przy uwierzytelnianiu.
- Ponieważ program DB2 UDB zapisuje identyfikatory użytkownika wielkimi literami, wartości 'N' i 'U' są sobie logicznie równoważne.
- Ustawienie opcji FOLD_PW nie ma znaczenia, gdy dla hasła wybrano ustawienie 'N'. Ponieważ hasło nie jest przesyłane, wielkość znaków nie gra roli.
- Dla obu tych opcji nie są zalecane wartości puste. Wartość pusta może być użyteczna, ponieważ powoduje, że program DB2 UDB będzie podejmował wiele prób odgadnięcia właściwego identyfikatora i hasła. Odbędzie się to jednak kosztem wydajności (może się zdarzyć, że pomyślne uwierzytelnienie w źródle danych będzie wymagało aż czterokrotnego przesłania identyfikatora i hasła).

Pojęcia pokrewne:

- “Cechy serwera wpływające na możliwości przekazywania” na stronie 136
- “Cechy serwera wpływające na globalną optymalizację” na stronie 147

Zadania pokrewne:

- “Registering server definitions for a data source” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

|

|

|

|

Informacje pokrewne:

- “DROP statement” w podręczniku *SQL Reference, Volume 2*
- “ALTER SERVER statement” w podręczniku *SQL Reference, Volume 2*
- “CREATE SERVER statement” w podręczniku *SQL Reference, Volume 2*

Rozdział 22. Opcje odwzorowań użytkowników w systemach stowarzyszonych

Opcje te są poprawne dla wszystkich relacyjnych źródeł danych. Opcje REMOTE_AUTHID i REMOTE_PASSWORD są ponadto poprawne dla następujących nierelacyjnych źródeł danych: BioRS, Documentum, Extended Search i usługi WWW. Opcja GUEST jest poprawna dla źródła danych BioRS.

Opcje te są używane w instrukcjach CREATE USER MAPPING i ALTER USER MAPPING.

Tabela 26. Opcje odwzorowania użytkowników i ich ustawienia

Opcja	Poprawne ustawienia	Ustawienie domyślne
ACCOUNTING	DRDA: Służy do określenia łańcucha rozliczeniowego DRDA. Dopuszczalna wartość to dowolny łańcuch o długości 255 znaków lub mniejszej. Ta opcja jest wymagana tylko w sytuacji, gdy wymagane jest przekazywanie danych rozliczeniowych. Więcej informacji zawiera podręcznik DB2 Connect - Podręcznik użytkownika.	Brak
GUEST	Określa, czy opakowanie ma korzystać z dostępu do serwera BioRS w trybie gościa. Y Opakowanie korzysta z dostępu do serwera BioRS w trybie gościa. N Opakowanie nie korzysta z dostępu do serwera BioRS w trybie gościa. Gdy tej opcji zostanie nadana wartość Y, będzie się ona wykluczać z opcjami REMOTE_AUTHID i REMOTE_PASSWORD.	N
REMOTE_AUTHID	Określa identyfikator autoryzowanego użytkownika używany w źródle danych. Dopuszczalna wartość to dowolny łańcuch o długości 255 znaków lub mniejszej.	Identyfikator autoryzowanego użytkownika używany do połączeń z programem DB2 Universal Database.
REMOTE_DOMAIN	Documentum: Określa domenę Windows NT używaną do uwierzytelniania użytkowników nawiązujących połączenia ze źródłem danych Documentum. Dopuszczalnym ustawieniem jest dowolna nazwa domeny Windows NT.	Domyślne uwierzytelnianie w domenie dla bazy danych Documentum.
REMOTE_PASSWORD	Określa hasło autoryzowanego użytkownika używane w źródle danych. Dopuszczalna wartość to dowolny łańcuch o długości 32 znaków lub mniejszej. Nie ma potrzeby określania tej opcji, gdy spełnione są poniższe warunki: <ul style="list-style-type: none"> • parametr konfiguracyjny menedżera bazy danych AUTHENTICATON ma wartość SERVER; • podczas połączenia z bazą danych DB2 podano identyfikator autoryzowanego użytkownika i hasło. Jeśli serwer wymaga podania hasła, a nie określi się tej opcji, obydwa powyższe warunki muszą być spełnione, w przeciwnym razie połączenie nie powiedzie się.	Hasło używane do połączeń z programem DB2 Universal Database, gdy obydwa warunki wymienione w kolumnie poprawnych ustawień są spełnione.

Pojęcia pokrewne:

- “Program DB2 Connect i architektura DRDA” w podręczniku *IBM DB2 Connect Podręcznik użytkownika*
- “Architektura DRDA i dostęp do danych” w podręczniku *IBM DB2 Connect Podręcznik użytkownika*

Zadania pokrewne:

- “Registering user mappings for a data source” w podręczniku *IBM DB2 Information Integrator Data Source Configuration Guide*

Rozdział 23. Opcje pseudonimu dla systemów stowarzyszonych

Tabela 27 i Tabela 28 zawierają listy opcji pseudonimu dla każdego źródła danych. Tabela 29 na stronie 244 zawiera opis każdej opcji pseudonimu oraz poprawne wartości i wartości domyślne.

Tabela 27. Dostępne opcje pseudonimów – od A do P

Źródło danych	ALL_VERSIONS	APPLICATIONID	BUSOBJ_NAME	CATEGORY	COLUMN_DELIMITER	DATASOURCE	DIRECTORY_PATH	FILE_PATH	FOLDERS	HMMTYPE	INSTANCE_PARSE_TIME	IS_REG_TABLE	KEY_COLUMN	MAXHIT	NAMESPACES	NEXT_TIME	PARENT	PROCESSORS
BioRS																		
BLAST						X												X
Documentum	X								X			X						
Entrez																	X	
Excel								X										
Extended Search		X		X	X								X					
HMMER					X					X								
Pliki o strukturze tabeli					X			X					X					
Usługi WWW															X			
WebSphere Business Integration			X												X			
XML							X	X			X					X		

Tabela 28 zawiera listę opcji pseudonimów dla każdego źródła danych, od R do X.

Tabela 28. Dostępne opcje pseudonimów – od R do X

Źródło danych	RANGE	REMOTE_OBJECT	SOAPACTION	SORTED	SORTFIELD	SORTORDER	STREAMING	TEMPLATE	TOTALMAXHIT	TIMEOUT	URL	VALIDATE	VALIDATE_DATA_FILE	VERTICAL_TABLE	XPATH	XPATH_EVAL_TIME
BioRS		X														
BLAST										X						
Documentum		X														
Entrez		X														
Excel	X															

Tabela 28. Dostępne opcje pseudonimów – od R do X (kontynuacja)

Źródło danych	RANGE	REMOTE_OBJECT	SOAPACTION	SORTED	SORTFIELD	SORTORDER	STREAMING	TEMPLATE	TOTALMAXHIT	TIMEOUT	URL	VALIDATE	VALIDATE_DATA_FILE	VERTICAL_TABLE	XPATH	XPATH_EVAL_TIME
Extended Search					X	X			X	X				X		
HMMER										X						
Pliki o strukturze tabeli				X									X			
Usługi WWW			X				X	X			X				X	
WebSphere Business Integration								X							X	
XML							X					X			X	X

Tabela 29 zawiera opis każdej opcji pseudonimu oraz poprawne wartości i wartości domyślne.

Tabela 29. Opcje pseudonimów i ich ustawienia

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
ALL_VERSIONS	Określa, czy mają być przeszukiwane wszystkie wersje obiektów. Poprawne wartości to y, Y, n i N. Wartość domyślna N oznacza, że podczas przetwarzania zapytania uwzględniane są tylko bieżące wersje obiektów. Ta opcja jest niepoprawna, gdy opcja IS_REG_TABLE ma wartość 'Y'.	N
APPLICATIONID	Określa nazwę aplikacji Extended Search, która ma być użyta do wyszukiwania. Nazwa ta musi istnieć w konfiguracyjnej bazie danych Extended Search. Ta opcja jest wymagana.	
BUSOBJ_NAME	Nazwa pliku definicji schematu XML (.xsd), reprezentującego obiekt biznesowy. Na przykład sap_bapi_customer_get_detail2. Ta opcja musi być podana w pseudonimie nadrzędnym.	
CATEGORY	Określa kategorie Extended Search, które mają być przeszukiwane. Jeśli opcja ta zostanie pominięta, należy podać co najmniej jedną nazwę źródła danych. Aby podać kilka kategorii, należy je rozdzielić średnikiem.	

Tabela 29. Opcje pseudonimów i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
COLUMN_DELIMITER	<p>Ujęty w cudzysłowy separator używany do rozdzielania kolumn pliku o strukturze tabeli. Separator kolumn może mieć więcej niż jeden znak. Jeśli nie zdefiniuje się separatora kolumn, domyślnym separatorem jest przecinek. Nie można użyć jako separatora pojedynczego znaku cudzysłowu. Separator kolumn musi być taki sam w całym pliku. Wartość pusta jest reprezentowana przez dwa separatory obok siebie lub separator i znak końca wiersza, gdy pole z wartością pustą występuje jako ostatnie w wierszu. Separator kolumn nie może występować w poprawnych danych kolumny.</p>	Separatorem domyślnym jest przecinek.
DATASOURCE	<p>Dla źródeł danych Extended Search: określa źródła danych Extended Search, które mają być przeszukiwane. Jeśli opcja ta zostanie pominięta, należy podać co najmniej jedną nazwę kategorii. Aby podać kilka źródeł danych, należy rozdzielić ich nazwy średnikiem.</p> <p>Dla źródła danych BLAST: nazwa źródła danych, dla którego będzie uruchamiane wyszukiwanie BLAST. Użyty dla tej opcji łańcuch musi znajdować się w pliku konfiguracyjnym demona BLAST. Ta opcja jest wymagana.</p> <p>Dla źródeł danych HMMER (typu PFAM): nazwa bazy danych HMM Profile, która ma być przeszukiwana przez program HMMPFAM. Łańcuch użyty w tej opcji musi się znajdować w pliku konfiguracyjnym demona HMMER. Ta opcja jest wymagana.</p> <p>Dla źródeł danych HMMER (typu SEARCH): nazwa pliku sekwencyjnego, który ma być przeszukiwany przez program HMMSEARCH. Łańcuch użyty w tej opcji musi się znajdować w pliku konfiguracyjnym demona HMMER. Ta opcja jest wymagana.</p>	
DIRECTORY_PATH	<p>Określa nazwę ścieżki do katalogu, który zawiera pliki XML. Przy użyciu tej opcji można utworzyć pojedynczy pseudonim dla wielu plików źródłowych XML. Opakowanie XML korzysta tylko z plików z rozszerzeniem .xml, które znajdują się w podanym katalogu. Opakowanie XML ignoruje wszystkie inne pliki w tym katalogu. Jeśli ta opcja pseudonimu zostanie podana, nie należy określać kolumny DOCUMENT. Opcja ta jest akceptowana tylko dla pseudonimu głównego (pseudonimu identyfikującego elementy najwyższego poziomu dokumentu XML).</p>	

Tabela 29. Opcje pseudonimów i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
FILE_PATH	<p>Dla źródeł danych Microsoft Excel: określa pełną ścieżkę do katalogu i nazwę pliku arkusza programu Excel, do którego ma być uzyskiwany dostęp. Ta opcja jest wymagana.</p> <p>Dla plików o strukturze tabeli: pełna ścieżka do pliku o strukturze tabeli, który ma być używany, ujęta w pojedyncze znaki cudzysłowu. Plik danych musi być plikiem standardowym lub dowiązaniem symbolicznym, nie może to być potok ani inny plik niestandardowego typu. Jeśli nie zostanie podana opcja FILE_PATH, należy podać opcję kolumny pseudonimu DOCUMENT. W przeciwnym razie nie można określić opcji kolumny pseudonimu DOCUMENT.</p> <p>Dla źródeł danych XML: określa ścieżkę do pliku dokumentu XML. Jeśli ta opcja pseudonimu zostanie podana, nie należy określać kolumny DOCUMENT. Opcja ta jest akceptowana tylko dla pseudonimu głównego (pseudonimu identyfikującego elementy najwyższego poziomu dokumentu XML).</p>	
FOLDERS	<p>Określa łańcuch zawierający logicznie połączone i poprawne składniowo predykaty FOLDER dla źródeł danych Documentum. Określenie predykatów FOLDER ogranicza zestaw dokumentów reprezentowanych przez ten pseudonim do dokumentów we wskazanych folderach.</p> <p>Podczas określania opcji FOLDERS należy całą jej wartość ująć w pojedyncze cudzysłowy, a w łańcuchu zastąpić pojedyncze cudzysłowy podwójnymi.</p> <p>Ta opcja jest niepoprawna, gdy opcja IS_REG_TABLE ma wartość 'Y'.</p>	
HMMTYPE	<p>Opcjonalnie: Alfabet używany zarówno w modelach, jak i w sekwencjach genów. Dozwolone wartości to NUCLEIC lub PROTEIN; wielkość liter w tych wartościach nie jest rozróżniana.</p>	PROTEIN
INSTANCE_PARSE_TIME	<p>Określa czas (w milisekundach) analizowania danych w jednym wierszu źródłowego dokumentu XML. W celu optymalizacji zapytań na dużych lub złożonych strukturach XML można zmodyfikować opcje INSTANCE_PARSE_TIME, XPATH_EVAL_TIME i NEXT_TIME. Opcja ta jest akceptowana tylko dla kolumn pseudonimu głównego (pseudonimu identyfikującego elementy najwyższego poziomu dokumentu XML). Wartość tej opcji może być liczbą całkowitą lub dziesiętną.</p>	7

Tabela 29. Opcje pseudonimów i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
IS_REG_TABLE	<p>Informuje, czy obiekt określony w opcji REMOTE_OBJECT jest zarejestrowaną tabelą Documentum. Poprawne wartości to 'y', 'Y', 'n' i 'N'.</p> <p>Nie można zmienić pseudonimu z obiektu Documentum na zarejestrowaną tabelę (ani na odwrót), zmieniając wartość tej opcji przy użyciu instrukcji ALTER NICKNAME. Aby to zrobić, należy usunąć pseudonim i utworzyć go ponownie.</p>	N
KEY_COLUMN	<p>Nazwa kolumny pliku tworzącej klucz, według którego ten plik jest sortowany, ujęta w pojedyncze cudzysłowy. Opcji tej należy używać wyłącznie dla plików sortowanych. Kolumna wskazywana przez opcję kolumny pseudonimu DOCUMENT nie może być użyta jako kolumna klucza.</p> <p>Obsługiwane są tylko klucze jednokolumnowe. Klucze wielokolumnowe są niedozwolone. Wartością opcji musi być nazwa kolumny zdefiniowanej w instrukcji CREATE NICKNAME. Kolumna musi być sortowana w porządku rosnącym. Kolumna klucza nie może zawierać wartości pustych: w jej definicji w instrukcji tworzenia pseudonimu musi być dodana opcja NOT NULL.</p> <p>W tej opcji rozróżniana jest wielkość liter. Jednak jeśli nazwa kolumny nie została zdefiniowana z podwójnymi cudzysłowami, program DB2 UDB zmieni wszystkie litery w tej nazwie na wielkie.</p>	Jeśli dla posortowanego pseudonimu wartość nie zostanie podana, zostanie przyjęta nazwa pierwszej kolumny pliku źródłowego pseudonimu.
MAXHIT	Liczba typu INTEGER określająca maksymalną liczbę wyników zwracanych w operacji wyszukiwania z każdego źródła.	50
NAMESPACES	<p>Przestrzenie nazw skojarzone z przedrostkami przestrzeni nazw używanymi w opcjach XPATH i TEMPLATE dla każdej kolumny. Służy do tego poniższa składnia:</p> <pre>NAMESPACES 'przedrostek1= "rzeczywista_przestrzeń_nazw1", przedrostek2= "rzeczywista_przestrzeń_nazw2" '</pre> <p>Nazwy poszczególnych przestrzeni nazw należy oddzielać przecinkami. Na przykład:</p> <pre>NAMESPACES ' c="http://www.myweb.com/cust", i="http://www.myweb.com/cust/id", n="http://www.myweb.com/cust/name"'</pre>	
NEXT_TIME	Określa czas (w milisekundach) wymagany do znalezienia kolejnych elementów źródłowych na podstawie wyrażenia XPath. W celu optymalizacji zapytań na dużych lub złożonych strukturach źródeł XML można zmodyfikować opcje NEXT_TIME, XPATH_EVAL_TIME i INSTANCE_PARSE_TIME. Ta opcja jest akceptowana zarówno dla pseudonimów głównych, jak i pozostałych.	1

Tabela 29. Opcje pseudonimów i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
PARENT	Opcję tę określa się tylko dla pseudonimu podrzędnego, dla którego nazwę pseudonimu nadrzędnego zmieniono za pomocą opcji REMOTE_OBJECT. Opcja PARENT służy do kojarzenia pseudonimów podrzędnych z nadrzędnymi, gdy w ramach schematu bazy danych DB2 zdefiniowano wiele rodzin pseudonimów. W tej nazwie rozróżniana jest wielkość liter.	
PROCESSORS	Określa liczbę procesorów używanych podczas obliczania zapytania BLAST. Opcja ta odpowiada opcji blastall -a.	1
RANGE	Określa zakres komórek źródła danych, które mają być użyte.	
REMOTE_OBJECT	<p>Dla źródeł danych BioRS: nazwa banku danych BioRS skojarzonego z pseudonimem. Ta nazwa określa schemat i bank danych BioRS dla pseudonimu. Nazwa ta określa także związki pseudonimu z innymi pseudonimami. Tryb rozróżniania wielkości liter w tej nazwie zależy od ustawień serwera BioRS i od wartości opcji serwera CASE_SENSITIVE. Do zmiany lub usunięcia tej nazwy nie można użyć instrukcji ALTER NICKNAME. Jeśli nazwa banku danych BioRS używana w tej opcji zmieni się, należy usunąć pseudonim i utworzyć go ponownie.</p> <p>Dla źródeł danych Documentum: nazwa typu obiektu Documentum skojarzonego z pseudonimem. Może to być dowolny typ obiektu Documentum lub zarejestrowana tabela. Nazwa zarejestrowanej tabeli musi być poprzedzona przedrostkiem z nazwą właściciela tabeli. Gdy zarejestrowana tabela należy do właściciela bazy danych Docbase, jako nazwy właściciela można użyć wartości dm_dbo. Ta opcja jest wymagana. Użycie opcji ALTER NICKNAME do zmiany wartości opcji REMOTE_OBJECT spowoduje błędy, gdy struktura nowego obiektu nie jest podobna do struktury pierwotnego obiektu.</p> <p>Dla źródeł danych Entrez: nazwa typu obiektu Entrez skojarzonego z pseudonimem. Nazwa ta określa schemat i bazę danych NCBI dla pseudonimu oraz jego związki z innymi pseudonimami. W tej nazwie nie jest rozróżniana wielkość liter.</p>	
SOAPACTION	Atrybut URI SOAPACTION z formatu WSDL (Web Service Description Language). Ta opcja jest wymagana dla pseudonimu głównego. Nie można jej użyć dla pseudonimów innych niż główne.	

Tabela 29. Opcje pseudonimów i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
SORTED	<p>Określa, czy źródło danych jest posortowane. Poprawne wartości to Y, y, n oraz N.</p> <p>Posortowane źródła danych muszą być sortowane w porządku rosnącym zgodnie z kolejnością zestawiania dla bieżących ustawień narodowych zdefiniowaną w ustawieniach w kategorii obsługi języków narodowych LC_COLLATE.</p> <p>Po określeniu, że źródło danych jest posortowane, należy nadać opcji VALIDATE_DATA_FILE wartość Y.</p>	N
SORTFIELD	<p>Określa nazwę pola, według którego mają być sortowane wyniki wyszukiwania. Wartość domyślna DOC_RANK to pole używane przez program Extended Search do określania stopnia dopasowania znalezionej dokumentu do kryteriów wyszukiwania. Można podać nazwę innego pola, ale musi ono istnieć w przeszukiwanym źródle.</p>	DOC_RANK
SORTORDER	<p>Określa kolejność sortowania zwracanych wyników; dostępne wartości to A (rosnący - ascending) lub D (malejący - descending).</p>	A
STREAMING	<p>Określa, czy dokument źródłowy XML ma być przed przetworzeniem dzielony na logiczne fragmenty. Fragmenty te odpowiadają węzłowi zgodnemu z wyrażeniem XPath pseudonimu. Następnie opakowanie analizuje i przetwarza źródło danych XML, fragment po fragmencie. Taki typ analizy minimalizuje wykorzystanie pamięci. Opcję tę określa się tylko dla pseudonimów głównych.</p> <p>Przetwarzanie strumieniowe można określić dla dowolnego dokumentu źródłowego XML (FILE, DIRECTORY, URI lub COLUMN). Opcja ta jest akceptowana tylko dla kolumn pseudonimu głównego (pseudonimu identyfikującego elementy najwyższego poziomu dokumentu XML).</p> <p>Poprawne wartości to:</p> <p>Y Dokumenty XML są analizowane.</p> <p>N Dokumenty XML nie są analizowane.</p> <p>Nie należy nadawać parametrowi STREAMING wartości YES, jeśli parametr VALIDATE ma wartość YES. Jeśli obydwa parametry będą miały wartość YES, zostanie wyświetlony komunikat o błędzie.</p>	N
TEMPLATE	<p>Dla źródeł danych WebSphere Business Integration: fragment szablonu pseudonimu, który ma być używany do konstruowania dokumentu wejściowego XML. Fragment musi być zgodny ze składnią podanego szablonu.</p> <p>Dla Usług WWW: fragment szablonu pseudonimu, który ma być używany do konstruowania żądania SOAP. Fragment musi być zgodny ze składnią podanego szablonu.</p>	

Tabela 29. Opcje pseudonimów i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
TOTALMAXHIT	Liczba typu INTEGER określająca maksymalną liczbę wyników zwracanych w operacji wyszukiwania ze wszystkich przeszukiwanych źródeł. Opakowanie złoży te wyniki w jedną tabelę wyników.	50
TIMEOUT	Dla źródeł danych Extended Search: liczba typu INTEGER określająca w sekundach czas oczekiwania na odpowiedź serwera przed przeterminowaniem żądania. Dla źródeł danych BLAST i HMMER: maksymalny czas w minutach, przez który opakowanie czeka na wyniki z demona.	Dla źródeł danych Extended Search: 30. Dla źródeł danych BLAST i HMMER: 60.
URL	Adres URL punktu końcowego usługi WWW. Ta opcja jest wymagana dla pseudonimu głównego. Nie można jej użyć dla pseudonimów innych niż główne. Obsługiwane są protokoły HTTP i HTTPS.	
VALIDATE	Określa, czy przed wyodrębnieniem z dokumentu danych XML ma być sprawdzana poprawność dokumentu źródłowego XML. Jeśli ta opcja ma wartość YES, następuje sprawdzenie, czy struktura dokumentu głównego jest zgodna ze schematem XML lub z definicją typu dokumentu (DTD). Opcja ta jest akceptowana tylko dla kolumn pseudonimu głównego (pseudonimu identyfikującego elementy najwyższego poziomu dokumentu XML). Poprawność dokumentu źródłowego XML nie będzie sprawdzana, gdy opakowanie XML nie znajdzie pliku schematu XML ani pliku DTD (.xsd lub .dtd). Jeśli poprawność nie będzie sprawdzana, program DB2 UDB nie wygeneruje komunikatu o błędzie. Dlatego należy upewnić się, że plik schematu XML lub plik DTD znajduje się w miejscu określonym w dokumencie źródłowym XML. Nie należy nadawać parametrowi VALIDATE wartości YES, jeśli parametr STREAMING ma wartość YES. Jeśli obydwa parametry będą miały wartość YES, zostanie wyświetlony komunikat o błędzie.	NO
VALIDATE_DATA_FILE	W przypadku plików sortowanych opcja ta włącza lub wyłącza sprawdzanie przez opakowanie, czy kolumna klucza jest posortowana w porządku rosnącym i czy występują puste wartości klucza. Jedyne poprawne wartości tej opcji to Y i N. Sprawdzanie odbywa się jeden raz w czasie rejestracji. Opcja ta jest niedozwolona, jeśli jako ścieżki do pliku użyto opcji kolumny pseudonimu DOCUMENT.	N
VERTICAL_TABLE	Określa format prezentacji wyników wyszukiwania. Jeśli opcja ta ma wartość YES, program Extended Search zwróci, oprócz kolumn zdefiniowanych przez użytkownika, również wszystkie pola skonfigurowane do zwrotu. Opakowanie przechowuje wyniki w tabeli pseudonimu jako pionową listę nazw kolumn.	NO

Tabela 29. Opcje pseudonimów i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
XPATH	<p>Określa wyrażenie XPATH identyfikujące elementy, które reprezentują poszczególne krotki. Opcja XPATH dla pseudonimu podrzędnego jest wyliczana w kontekście ścieżki określonej przez opcję XPATH dla jej pseudonimu nadrzędnego. To wyrażenie XPATH służy jako kontekst do wyliczania wartości kolumn identyfikowanych przez opcje kolumn pseudonimu XPATH.</p> <p>Dla źródeł danych XML: w wyrażeniu XPATH nie należy określać przedrostka przestrzeni nazw. Opakowanie XML nie obsługuje przestrzeni nazw.</p>	
XPATH_EVAL_TIME	<p>Określa czas (w milisekundach) wyliczania wyrażenia XPath dla pseudonimu i znalezienia pierwszego elementu. W celu optymalizacji zapytań dotyczących dużych lub złożonych struktur źródeł danych XML można zmodyfikować opcje XPATH_EVAL_TIME, INSTANCE_PARSE_TIME i NEXT_TIME. Ta opcja jest akceptowana zarówno dla pseudonimów głównych, jak i pozostałych. Wartość tej opcji może być liczbą całkowitą lub dziesiętną.</p>	1

Rozdział 24. Opcje kolumn pseudonimu dla systemów stowarzyszonych

Informacje o kolumnach można zdefiniować za pośrednictwem instrukcji CREATE NICKNAME lub ALTER NICKNAME przy użyciu parametrów nazywanych opcjami kolumn pseudonimu.

Poniższa tabela zawiera listę opcji kolumn pseudonimu dla każdego źródła danych.

Tabela 30. Dostępne opcje kolumn pseudonimu.

Źródło danych	ALL_VALUES	DEFAULT	DELIMITER	DOCUMENT	ESCAPE_INPUT	FOREIGN_KEY	INDEX	IS_REPEATING	NUMERIC_STRING	PRIMARY_KEY	REMOTE_NAME	SOAPACTIONCOLUMN	TEMPLATE	URLCOLUMN	VARCHAR_NO_TRAILING_BLANKS	XPATH
BLAST		X	X				X									
DB2 Universal Database for iSeries,									X							
DB2 Universal Database for z/OS and OS/390									X							
DB2 Universal Database for VM and VSE									X							
DB2 Universal Database dla systemów Linux, UNIX i Windows									X							
Documentum	X		X					X			X					
Informix									X							
Microsoft SQL Server									X							
ODBC									X							
OLE DB																
Oracle									X						X	
Sybase									X							
Pliki o strukturze tabeli				X												
Teradata									X							
WebSphere Business Integration					X	X				X			X			X

Tabela 30. Dostępne opcje kolumn pseudonimu. (kontynuacja)

Źródło danych	ALL_VALUES	DEFAULT	DELIMITER	DOCUMENT	ESCAPE_INPUT	FOREIGN_KEY	INDEX	IS_REPEATING	NUMERIC_STRING	PRIMARY_KEY	REMOTE_NAME	SOAPACTIONCOLUMN	TEMPLATE	URLCOLUMN	VARCHAR_NO_TRAILING_BLANKS	XPATH
Usługi WWW					X	X				X		X	X	X		X
XML				X		X				X						X

Tabela 31. Opcje kolumn i ich ustawienia

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
ALL_VALUES	Powoduje, że zostaną zwrócone wszystkie wartości powtarzającego się atrybutu rozdzielone określonym separatorem. W przypadku braku tej opcji lub gdy ma ona wartość N, zwrócona zostanie tylko ostatnia wartość powtarzającego się atrybutu. Opcja ALL_VALUES może być określona tylko dla kolumn typu VARCHAR, dla których opcja IS_REPEATING ma wartość 'Y' (i jest niepoprawna, gdy IS_REG_TABLE = 'Y').	
DEFAULT	Określa nową wartość domyślną dla następujących ustalonych kolumn wejściowych: <ul style="list-style-type: none"> E_value QueryStrands GapAlign NMismatchPenalty NMatchReward Matrix FilterSequence NumberOfAlignments GapCost ExtendedGapCost WordSize ThresholdEx <p>Ta nowa wartość przesłania wstępnie zdefiniowaną wartość domyślną. Nowa wartość domyślna musi być tego samego typu, co wartość wskazana dla danej kolumny.</p>	

Tabela 31. Opcje kolumn i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
DELIMITER	<p>Dla źródeł danych Documentum: określa łańcuch separatora, który ma być używany, gdy konkatelowanych jest wiele wartości powtarzalnego atrybutu. Separator może składać się z kilku znaków. Ta opcja jest poprawna tylko dla atrybutów obiektów z typem danych VARCHAR, dla których opcja IS_REPEATING ma wartość Y.</p> <p>Dla źródeł danych BLAST: znaki separatora, które mają być używane w celu określenia punktu końcowego informacji wiersza definicji dla kolumny, dla której stosowana jest ta opcja. Jeśli wartość tej opcji ma kilka znaków, pierwsze wystąpienie dowolnego z tych znaków sygnalizuje koniec informacji w tym polu. Wartością domyślną jest znak końca wiersza. Opcja ta jest wymagana, chyba że ostatnia określona kolumna ma zawierać pozostałą część wiersza definicji.</p>	<p>Dla źródeł danych Documentum: separatorem domyślnym jest przecinek.</p> <p>Dla źródeł danych BLAST: separatorem domyślnym jest znak końca wiersza.</p>

Tabela 31. Opcje kolumn i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
DOCUMENT	<p>Dla plików o strukturze tabeli: określa rodzaj pliku o strukturze tabeli. To opakowanie obsługuje tylko wartość FILE dla tej opcji. Dla każdego pseudonimu opcję DOCUMENT można określić tylko dla jednej kolumny. Kolumna skojarzona z opcją DOCUMENT musi zawierać dane typu VARCHAR lub CHAR.</p> <p>Użycie opcji kolumny pseudonimu DOCUMENT zamiast opcji pseudonimu FILE_PATH powoduje, że w momencie uruchomienia zapytania zostanie dostarczony plik odpowiadający temu pseudonimowi. Gdy opcja DOCUMENT ma wartość FILE, podczas uruchamiania zapytania dostarczana jest pełna ścieżka pliku, którego schemat odpowiada definicji tego pseudonimu.</p> <p>Dla XML: określa, że dana kolumna jest kolumną typu DOCUMENT. Wartość kolumny DOCUMENT wskazuje na typ źródła danych XML dostarczany do pseudonimu w momencie uruchomienia zapytania. Opcja ta jest akceptowana tylko dla kolumn pseudonimu głównego (pseudonimu identyfikującego elementy najwyższego poziomu dokumentu XML). Dla każdego pseudonimu opcję DOCUMENT można określić tylko dla jednej kolumny. Kolumna skojarzona z opcją DOCUMENT musi zawierać dane typu VARCHAR.</p> <p>Użycie opcji kolumny pseudonimu DOCUMENT zamiast opcji pseudonimu FILE_PATH lub DIRECTORY_PATH powoduje, że w momencie uruchomienia zapytania zostanie dostarczony dokument odpowiadający temu pseudonimowi.</p> <p>Poprawne wartości opcji DOCUMENT to:</p> <p>FILE Określa, że wartość kolumny pseudonimu jest powiązana z nazwą ścieżki do pliku. Po uruchomieniu zapytania dostarczane są dane z tego pliku.</p> <p>DIRECTORY Określa, że wartość kolumny pseudonimu jest powiązana z nazwą ścieżki do katalogu, który zawiera wiele plików danych XML. Po uruchomieniu zapytania dostarczane są dane XML z wielu plików. Dane znajdują się w plikach XML w określonej ścieżce do katalogu. Opakowanie XML korzysta tylko z plików z rozszerzeniem .xml, które znajdują się w podanym katalogu. Opakowanie XML ignoruje wszystkie inne pliki w tym katalogu.</p> <p>URI Określa, że wartość kolumny pseudonimu jest powiązana z nazwą ścieżki do zdalnego pliku XML, do którego odwołuje się identyfikator URI. Adres URI wskazuje zdalne położenie tego pliku XML w sieci WWW.</p> <p>COLUMN Określa, że dokument XML jest przechowywany w kolumnie relacyjnej.</p>	
ELEMENT_NAME	Określa nazwę elementu BioRS. Tryb rozróżniania wielkości liter w tej nazwie zależy od ustawień serwera BioRS i od wartości opcji serwera CASE_SENSITIVE. Nazwę elementu BioRS należy podać tylko wtedy, gdy jest ona inna niż nazwa kolumny.	

Tabela 31. Opcje kolumn i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
ESCAPE_INPUT	<p>Określa, czy znaki specjalne XML w wartościach wejściowych XML mają być zastępowane. Należy użyć tej opcji, aby włączyć fragmenty XML jako dane wejściowe z powtarzającymi się elementami. Dla kolumn, dla których ustawiono opcję kolumny ESCAPE_INPUT, należy zdefiniować również opcję kolumny TEMPLATE. Typem danych kolumny musi być VARCHAR lub CHAR.</p> <p>Poprawne wartości to:</p> <p>Y Gdy dane wejściowe XML zawierają znaki specjalne, są one zastępowane przez ich odpowiedniki używane w języku XML do reprezentowania znaków wejściowych.</p> <p>N Znaki wejściowe są zachowywane w postaci, w jakiej są wyświetlane.</p>	T
FOREIGN_KEY	<p>Wskazuje, że dany pseudonim jest pseudonimem podrzędnym i określa nazwę odpowiadającego mu pseudonimu nadrzędnego. Pseudonim może mieć co najwyżej jedną opcję kolumny FOREIGN_KEY. W wartości tej opcji jest rozróżniana wielkość liter. Tabela wskazywana przez tę opcję zawiera klucz wygenerowany przez opakowanie. Dla takiej kolumny nie można określić opcji XPATH. Kolumny tej można używać tylko do łączenia pseudonimów nadrzędnych z podrzędnymi.</p> <p>Wykonanie instrukcji CREATE NICKNAME z opcją FOREIGN_KEY nie powiedzie się, gdy pseudonim nadrzędny ma inną nazwę schematu.</p> <p>Jeśli nazwa pseudonimu, do którego odwołuje się klauzula FOREIGN_KEY, nie została w sposób jawny zdefiniowana małymi lub mieszanymi literami i ujęta w cudzysłowy w odpowiedniej instrukcji CREATE NICKNAME, w odwołaniu do tego pseudonimu w klauzuli FOREIGN_KEY konieczne jest użycie wielkich liter.</p> <p>Dla kolumny, dla której zostanie określona ta opcja, nie można określić żadnej innej opcji.</p>	
INDEX	Liczba porządkowa kolumny, dla której opcja ta występuje, w grupie kolumn wiersza definicji. Ta opcja jest wymagana.	
IS_INDEXED	Wskazuje, czy odpowiednia kolumna jest indeksowana (czy można się do niej odwoływać w predykcje). Poprawne wartości to Y i N. Wartość Y można określić tylko dla kolumn, dla których odpowiadający im element jest indeksowany przez serwer BioRS.	Podczas tworzenia pseudonimu opcja ta jest automatycznie dodawana - z wartością Y - do każdej kolumny odpowiadającej elementowi indeksowanemu przez serwer BioRS.
IS_REPEATING	<p>Wskazuje, czy dana kolumna jest wielowartościowa. Poprawne wartości to Y i N.</p> <p>Tylko ostatnia wartość jest zwracana dla:</p> <ul style="list-style-type: none"> • powtarzających się atrybutów typu innego niż VARCHAR, • kolumn typu VARCHAR z określoną opcją ALL_VALUES o wartości 'N'. <p>W celu obejścia tego ograniczenia można utworzyć podwójną definicję kolumny z powtarzającym się atrybutem.</p>	N

Tabela 31. Opcje kolumn i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
NUMERIC_STRING	Informuje, czy kolumna zawiera łańcuchy znaków numerycznych.	N
	<p>Y Taka kolumna zawiera łańcuchy znaków numerycznych '0', '1', '2', '9'. Kolumna nie zawiera spacji. Jeśli kolumna zawiera tylko łańcuchy numeryczne, a po nich znaki odstępu, nie należy podawać wartości Y.</p> <p>Nadanie opcji NUMERIC_STRING dla danej kolumny wartości Y informuje optymalizator, że ta kolumna nie zawiera znaków odstępu, które mogłyby zaburzyć porządek sortowania danych kolumny. Opcja ta jest przydatna w sytuacji, gdy kolejność zestawiania w źródle danych jest inna niż używana przez serwer stowarzyszony. Kolumny z tą opcją nie będą wykluczane z wartościowania zdalnego ze względu na inną kolejność zestawiania.</p> <p>N Taka kolumna zawiera dane typu innego niż łańcuchy znaków numerycznych, albo oprócz łańcuchów znaków numerycznych zawiera także znaki odstępu.</p>	
PRIMARY_KEY	<p>Wskazuje, że dany pseudonim jest pseudonimem nadrzędnym. Typem danych kolumny musi być VARCHAR (16). Pseudonim może mieć najwyżej jedną opcję kolumny PRIMARY_KEY. Jedyna poprawna wartość to YES. Kolumna wskazywana przez tę opcję zawiera klucz wygenerowany przez opakowanie. Dla takiej kolumny nie można określić opcji XPATH. Kolumny tej można używać tylko do łączenia pseudonimów nadrzędnych z podrzędnymi.</p> <p>Dla kolumny, dla której zostanie określona ta opcja, nie można określić żadnej innej opcji.</p>	
REFERENCED_OBJECT	Ta opcja jest poprawna tylko dla kolumn, dla których typem danych BioRS jest Reference. Opcja ta określa nazwę banku danych BioRS, do którego odwołuje się bieżąca kolumna. Tryb rozróżniania wielkości liter w tej nazwie zależy od ustawień serwera BioRS i od wartości opcji serwera CASE_SENSITIVE.	
REMOTE_NAME	Określa nazwę odpowiedniego atrybutu lub kolumny źródła danych Documentum. Opcja ta odwzorowuje nazwy zdalnych atrybutów lub kolumn na lokalne nazwy kolumn w programie DB2 UDB.	Nazwa kolumny w programie DB2 UDB.
SOAPACTIONCOLUMN	<p>Kolumna określa dynamicznie atrybut URI SOAPACTION na podstawie formatu WSDL. Opcję tę określa się tylko dla pseudonimów głównych.</p> <p>Dla kolumny, dla której zostanie określona ta opcja, nie można określić żadnej innej opcji.</p>	
TEMPLATE	Fragment szablonu kolumny, który ma być używany do konstruowania dokumentu wejściowego XML. Fragment musi być zgodny ze składnią podanego szablonu.	
URLCOLUMN	<p>Kolumna służąca do dynamicznego określania adresu URL dla punktu końcowego usługi WWW podczas uruchamiania zapytania. Opcję tę określa się tylko dla pseudonimów głównych.</p> <p>Dla kolumny, dla której zostanie określona ta opcja, nie można określić żadnej innej opcji.</p>	

Tabela 31. Opcje kolumn i ich ustawienia (kontynuacja)

Opcja	Opis i poprawne ustawienia	Ustawienie domyślne
VARCHAR_NO_ TRAILING_BLANKS	<p>Opcja ta ma zastosowanie do źródeł danych typu znakowego o zmiennej długości, w których długość nie jest dopełniana podczas porównywania odstępami końcowymi.</p> <p>Niektóre źródła danych, na przykład Oracle, nie mają semantyki porównywania danych znakowych dopełnianych spacjami dającej takie same wyniki, jak semantyka porównywania w programie DB2 dla systemów Linux, UNIX i Windows. Opcję tę należy ustawiać, gdy powinna odnosić się tylko do konkretnej kolumny typu VARCHAR lub VARCHAR2 w obiekcie źródła danych.</p> <p>Y W takich kolumnach typu VARCHAR nie ma końcowych znaków odstępów lub źródło danych ma podobną semantykę porównywania danych dopełnianych odstępami, jak serwer stowarzyszony.</p> <p>Serwer stowarzyszony przesyła operacje porównywania znaków do źródła danych w celu przetworzenia.</p> <p>N W takich kolumnach typu VARCHAR występują końcowe znaki odstępów lub źródło danych ma semantykę porównywania danych dopełnianych odstępami inną niż serwer stowarzyszony.</p> <p>Jeśli kompensacja do równoważnych semantyk jest niemożliwa, serwer stowarzyszony przetwarza operacje porównywania znaków. Na przykład przebudowuje predykat.</p>	N dla niezgodnych źródeł danych
XPATH	Określa wyrażenie XPath w dokumencie XML, który zawiera dane odpowiadające tej kolumnie. W opakowaniu wyrażenie XPath jest obliczane po zastosowaniu w instrukcji CREATE NICKNAME tego wyrażenia z opcji pseudonimu XPATH.	

Pojęcia pokrewne:

- “Analiza przekazywania do źródła” na stronie 135

Zadania pokrewne:

- “Globalna optymalizacja” na stronie 146

Rozdział 25. Opcje odwzorowań funkcji w systemach stowarzyszonych

W programie DB2 Information Integrator dostępne są domyślne odwzorowania między wbudowanymi funkcjami źródeł danych a wbudowanymi funkcjami programu DB2. W przypadku większości źródeł danych domyślne odwzorowania funkcji są zdefiniowane w opakowaniach. Aby móc skorzystać z funkcji źródła danych nierozpoznawanej przez serwer stowarzyszony, należy utworzyć odwzorowanie między funkcją źródła danych a odpowiadającą jej funkcją w stowarzyszonej bazie danych.

Podstawowa rola opcji odwzorowań funkcji polega na definiowaniu informacji o ewentualnym koszcie wykonywania funkcji w źródle danych. W drodze analizy przekazywania do źródła ustala się, czy funkcja obecna w źródle danych może służyć do realizacji odpowiedniej funkcji w zapytaniu. Optymalizator zapytań decyduje, czy przekazanie przetwarzania funkcji do źródła danych jest rozwiązaniem o najniższym koszcie.

Informacje statystyczne zamieszczone w definicji odwzorowań funkcji pomagają optymalizatorowi zapytań porównać szacowany koszt wykonania funkcji w źródle danych z szacowanym kosztem wykonania funkcji DB2.

Tabela 32. Opcje odwzorowań funkcji z ustawieniami

Opcja	Poprawne ustawienia	Ustawienie domyślne
DISABLE	Wyłączenie domyślnego odwzorowania funkcji. Dopuszczalne wartości to 'Y' (tak) i 'N' (nie).	'N'
INITIAL_INSTS	Szacowana liczba instrukcji przetwarzanych przy pierwszym i ostatnim wywołaniu funkcji w źródle danych.	'0'
INITIAL_IOS	Szacowana liczba operacji we/wy wykonywanych przy pierwszym i ostatnim wywołaniu funkcji w źródle danych.	'0'
IOS_PER_ARGBYTE	Szacowana liczba operacji we/wy realizowanych dla każdego bajtu zestawu danych przekazywanego jako argument funkcji źródła danych.	'0'
IOS_PER_INVOC	Szacowana liczba operacji we/wy realizowanych przy każdym wywołaniu funkcji w źródle danych.	'0'
INSTS_PER_ARGBYTE	Szacowana liczba przetwarzanych instrukcji dla każdego bajta zestawu danych przekazywanego jako argument funkcji źródła danych.	'0'
INSTS_PER_INVOC	Szacowana liczba instrukcji przetwarzanych przy każdym wywołaniu funkcji w źródle danych.	'450'
PERCENT_ARGBYTES	Szacowana średnia część procentowa argumentu (w bajtach), którą funkcja źródła danych faktycznie odczyta.	'100'
REMOTE_NAME	Nazwa funkcji źródła danych.	nazwa lokalna

Rozdział 26. Poprawne typy serwerów w instrukcjach SQL

Typ serwera wskazuje rodzaj źródła danych reprezentowanego przez definicję tego serwera. Typy serwerów zależne są od producenta, przeznaczenia i systemu operacyjnego. Dozwolone wartości zależą ponadto od używanego opakowania.

Dla większości źródeł danych poprawny typ serwera należy określić w instrukcji CREATE SERVER.

Opakowanie BioRS

Źródła danych BioRS.

Typ serwera	Źródło danych
Niewymagany w instrukcji CREATE SERVER.	BioRS

Opakowanie BLAST

Źródła danych BLAST obsługiwane przez demona BLAST.

Typ serwera	Źródło danych
BLASTN	Wyszukiwania BLAST, w których sekwencja nukleotydów jest porównywana z zawartością bazy danych sekwencji nukleotydów w celu znalezienia sekwencji z obszarami homologicznymi do obszarów oryginalnej sekwencji.
BLASTP	Wyszukiwania BLAST, w których sekwencja aminokwasów jest porównywana z zawartością bazy danych sekwencji aminokwasów w celu znalezienia sekwencji z obszarami homologicznymi do obszarów oryginalnej sekwencji.
BLASTX	Wyszukiwania BLAST, w których sekwencja nukleotydów jest porównywana z zawartością bazy danych sekwencji aminokwasów w celu znalezienia sekwencji z obszarami homologicznymi do obszarów oryginalnej sekwencji.
TBLASTN	Wyszukiwania BLAST, w których sekwencja aminokwasów jest porównywana z zawartością bazy danych sekwencji nukleotydów w celu znalezienia sekwencji z obszarami homologicznymi do obszarów oryginalnej sekwencji.
TBLASTX	Wyszukiwania BLAST, w których sekwencja nukleotydów jest porównywana z zawartością bazy danych sekwencji nukleotydów w celu znalezienia sekwencji z obszarami homologicznymi do obszarów oryginalnej sekwencji.

Opakowanie CTLIB

Źródła danych Sybase obsługiwane przez program klienta CTLIB.

Typ serwera	Źródło danych
SYBASE	Sybase

Opakowanie Documentum

Źródła danych Documentum obsługiwane przez interfejs API/bibliotekę klienta Documentum.

Typ serwera	Źródło danych
DCTM	Documentum

Opakowanie DRDA

Źródła danych z rodziny DB2

Tabela 33. DB2 dla systemów Linux, UNIX i Windows

Typ serwera	Źródło danych
DB2/UIDB	IBM DB2 Universal Database
DB2/6000	IBM DB2 for AIX
DB2/AIX	IBM DB2 for AIX
DB2/HPUX	IBM DB2 for HP-UX
DB2/HP	IBM DB2 for HP-UX
DB2/NT	IBM DB2 for Windows NT
DB2/EEE	IBM DB2 Enterprise-Extended Edition
DB2/SUN	IBM DB2 for Solaris
DB2/PE	IBM DB2 Personal Edition
DB2/2	IBM DB2 for OS/2
DB2/LINUX	IBM DB2 for Linux
DB2/PTX	IBM DB2 for NUMA-Q
DB2/SCO	IBM DB2 for SCO Unixware

Tabela 34. DB2 for iSeries (i AS/400)

Typ serwera	Źródło danych
DB2/400	IBM DB2 for iSeries and AS/400

Tabela 35. DB2 for z/OS and OS/390

Typ serwera	Źródło danych
DB2/ZOS	IBM DB2 for z/OS
DB2/390	IBM DB2 for OS/390
DB2/MVS	IBM DB2 for MVS

Tabela 36. DB2 Server for VM and VSE

Typ serwera	Źródło danych
DB2/VM	IBM DB2 for VM
DB2/VSE	IBM DB2 for VSE
SQL/DS	IBM SQL/DS

Opakowanie Entrez

Źródła danych Entrez.

Typ serwera	Źródło danych
NUCLEOTIDE	Entrez
PUBMED	Entrez

Opakowanie Excel

Źródła danych Excel obsługiwane przez programy Microsoft Excel 97, 2000 i 2002.

Typ serwera	Źródło danych
Niewymagany w instrukcji CREATE SERVER.	Microsoft Excel

Opakowanie Extended Search

Źródła danych Extended Search obsługiwane przez bibliotekę klienta Extended Search.

Typ serwera	Źródło danych
Niewymagany w instrukcji CREATE SERVER.	IBM Lotus Extended Search

Opakowanie HMMER

Źródła danych HMMER obsługiwane przez demona HMMER.

Typ serwera	Źródło danych
PFAM	HMMER
SEARCH	HMMER

Opakowanie Informix

Źródła danych Informix obsługiwane przez oprogramowanie SDK klienta Informix.

Typ serwera	Źródło danych
INFORMIX	Informix

Opakowanie MSSQLODBC3

Źródła danych Microsoft SQL Server obsługiwane przez sterownik DataDirect Connect ODBC 3.6 lub ODBC 3.0 (lub nowszy).

Typ serwera	Źródło danych
MSSQLSERVER	Microsoft SQL Server

Opakowanie NET8

Źródła danych Oracle obsługiwane przez oprogramowanie klienta Oracle NET8.

Typ serwera	Źródło danych
ORACLE	Oracle Version 8.0. lub nowszy

Opakowanie ODBC

Źródła danych ODBC obsługiwane przez sterownik ODBC 3.x.

Typ serwera	Źródło danych
ODBC	ODBC

Opakowanie OLE DB

Dostawcy OLE DB zgodni ze specyfikacją Microsoft OLE DB 2.0 lub nowszą.

Typ serwera	Źródło danych
Niewymagany w instrukcji CREATE SERVER.	Dowolny dostawca OLE DB

Opakowanie pliku o strukturze tabeli

Źródła danych w postaci plików o strukturze tabeli.

Typ serwera	Źródło danych
Niewymagany w instrukcji CREATE SERVER.	Pliki o strukturze tabeli

Opakowanie Teradata

Źródła danych Teradata obsługiwane przez oprogramowanie klienta Teradata V2R3, V2R4 i V2R5.

Typ serwera	Źródło danych
TERADATA	Teradata

Opakowanie usług WWW

Źródła danych usług WWW.

Typ serwera	Źródło danych
Niewymagany w instrukcji CREATE SERVER.	Dowolne źródło danych usług WWW.

Opakowanie WebSphere Business Integration

Źródła danych aplikacji biznesowych obsługiwanych przez opakowanie WeSphere Business Integration.

Typ serwera	Źródło danych
WBI	WebSphere Business Integration, wersja 2.2 lub 2.3

Opakowanie XML

Źródła danych XML.

Typ serwera	Źródło danych
Niewymagany w instrukcji CREATE SERVER.	XML

Rozdział 27. Domyślne proste odwzorowania typów

Istnieją dwa rodzaje odwzorowań typów między typami źródła danych a typami w stowarzyszonej bazie danych: odwzorowania proste i odwzorowania zwrotne. W *prostym odwzorowaniu typów* typy zdalne są odwzorowane na odpowiednie typy lokalne.

Można zmienić domyślne odwzorowanie typów lub utworzyć nowe odwzorowanie, korzystając z instrukcji CREATE TYPE MAPPING.

Odwzorowania te obowiązują we wszystkich obsługiwanych wersjach programów, chyba że zaznaczono inaczej.

Dla wszystkich domyślnych prostych odwzorowań typów danych w kierunku od źródła danych do bazy DB2 dla systemów Linux, UNIX i Windows schematem stowarzyszonym DB2 jest SYSIBM.

W poniższych tabelach przedstawiono domyślne proste odwzorowania typów danych między bazami DB2 dla systemów Linux, UNIX i Windows a typami w źródłach danych.

Źródła danych DB2 for z/OS and OS/390

Tabela 37. Domyślne proste odwzorowania typów danych dla źródeł DB2 for z/OS and OS/390 (nie wszystkie kolumny pokazano)

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	-	-	T	-	CHAR	-	0	T
CHAR	255	32672	-	-	T	-	VARCHAR	-	0	T
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-

Tabela 37. Domyślne proste odwzorowania typów danych dla źródeł DB2 for z/OS and OS/390 (nie wszystkie kolumny pokazano) (kontynuacja)

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
ROWID	-	-	-	-	T	-	VARCHAR	40	-	T
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	T	-	VARCHAR	-	0	T
VARG	1	16336	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	0	N

Źródła danych DB2 dla iSeries

Tabela 38. Domyślne proste odwzorowania typów danych dla źródeł DB2 dla iSeries (nie wszystkie kolumny pokazano)

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	-	-	T	-	CHAR	-	0	T
CHAR	255	32672	-	-	T	-	VARCHAR	-	0	T
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-

Tabela 38. Domyślne proste odwzorowania typów danych dla źródeł DB2 dla iSeries (nie wszystkie kolumny pokazano) (kontynuacja)

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
1 FLOAT	4	-	-	-	-	-	REAL	-	-	-
1 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	-	-	T	-	VARCHAR	-	0	T
VARG	1	16336	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	0	N

Źródła danych DB2 Server dla VM i VSE

Tabela 39. Domyślne proste odwzorowania typów danych dla źródeł DB2 Server dla VM i VSE (nie wszystkie kolumny pokazano)

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	-	-	T	-	CHAR	-	0	T
CLOB	-	-	-	-	-	-	CLOB	-	-	-

Tabela 39. Domyślne proste odwzorowania typów danych dla źródeł DB2 Server dla VM i VSE (nie wszystkie kolumny pokazano) (kontynuacja)

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
DATE	-	-	-	-	-	-	DATE	-	0	-
DBAHW	-	-	-	-	-	-	SMALLINT	-	0	-
DBAINT	-	-	-	-	-	-	INTEGER	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
I FLOAT	4	-	-	-	-	-	REAL	-	-	-
I 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	-	-	T	-	VARCHAR	-	0	T
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPH	1	16336	-	-	-	-	VARGRAPHIC	-	0	N

Źródła danych DB2 dla systemów Linux, UNIX i Windows

Tabela 40. Domyślne proste odwzorowania typów danych dla źródeł DB2 dla systemów Linux, UNIX i Windows (nie wszystkie kolumny pokazano)

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	-

Tabela 40. Domyślne proste odwzorowania typów danych dla źródeł DB2 dla systemów Linux, UNIX i Windows (nie wszystkie kolumny pokazano) (kontynuacja)

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	-	-	-	-	-	-	CHAR	-	0	N
CHAR	-	-	-	-	T	-	CHAR	-	0	T
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	-	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
LONGVAR	-	-	-	-	N	-	CLOB	-	-	-
LONGVAR	-	-	-	-	T	-	BLOB	-	-	-
LONGVARG	-	-	-	-	-	-	DBCLOB	-	-	-
REAL	-	-	-	-	-	-	REAL	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	0	N
VARCHAR	-	-	-	-	T	-	VARCHAR	-	0	T
VARGRAPH	-	-	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	0	N

Źródła danych Informix

Tabela 41. Domyślne proste odwzorowania typów danych dla źródeł Informix (nie wszystkie kolumny pokazano)

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	-	-
BOOLEAN	-	-	-	-	-	-	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	130	-	-	-	-	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	-	-	-
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	-	-	-

Tabela 41. Domyślne proste odwzorowania typów danych dla źródeł Informix (nie wszystkie kolumny pokazano) (kontynuacja)

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
-----------------	------------------	------------------	--------------------	--------------------	-----------------	-----------------------	--------------------	------------------	-----------------	--------------------

Uwagi:

- W przypadku typu Informix DATETIME serwer stowarzyszony DB2 dla UNIX i Windows używa kwalifikatora wysokopoziomowego Informix jako parametru REMOTE_LENGTH, a kwalifikatora niskopoziomowego Informix jako parametru REMOTE_SCALE.
- Kwalifikatory Informix są stałymi "TU_" zdefiniowanymi w pliku datatype.h za pomocą pakietu SDK klienta Informix. Stałe te, to:

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)	

Źródła danych Microsoft SQL Server

Tabela 42. Domyślne proste odwzorowania typów danych dla źródeł Microsoft SQL Server

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 ⁴	-	-	-	-	-	-	BIGINT	-	-	-
binary	1	254	-	-	-	-	CHARACTER	-	-	T
binary	255	8000	-	-	-	-	VARCHAR	-	-	T
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	-	-	-

Tabela 42. Domyślne proste odwzorowania typów danych dla źródeł Microsoft SQL Server (kontynuacja)

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
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	-	T
int	-	-	-	-	-	-	INTEGER	4	-	-
intn	-	-	-	-	-	-	INTEGER	4	-	-
money	-	-	-	-	-	-	DECIMAL	19	4	-
moneyn	-	-	-	-	-	-	DECIMAL	19	4	-
nchar	1	127	-	-	-	-	CHAR	-	-	N
nchar	128	4000	-	-	-	-	VARCHAR	-	-	N
numeric	1	31	0	31	-	-	DECIMAL	-	-	-
numeric	32	38	0	38	-	-	DOUBLE	8	-	-
numericn	32	38	0	38	-	-	DOUBLE	-	-	-
numericn	1	31	0	31	-	-	DECIMAL	-	-	-
ntext ²	-	-	-	-	-	-	CLOB	2147483647	-	T
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_BIGINT ⁴	-	-	-	-	-	-	BIGINT	-	-	-
SQL_BINARY	1	254	-	-	-	-	CHARACTER	-	-	T
SQL_BINARY	255	8000	-	-	-	-	VARCHAR	-	-	T
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N

Tabela 42. Domyślne proste odwzorowania typów danych dla źródeł Microsoft SQL Server (kontynuacja)

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_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	-	-
SQL_FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_GUID ²	1	4000	-	-	T	-	VARCHAR	16	-	T
SQL_INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
SQL_LONGVARCHAR	-	-	-	-	-	-	CLOB	2147483647	-	N
SQL_LONGVARBINARY	-	-	-	-	-	-	BLOB	-	-	T
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	-	-	T
SQL_VARCHAR	1	8000	-	-	-	-	VARCHAR	-	-	N
text	-	-	-	-	-	-	CLOB	-	-	N
timestamp	-	-	-	-	-	-	VARCHAR	8	-	T
tinyint	-	-	-	-	-	-	SMALLINT	2	-	-
uniqueidentifier ²	1	4000	-	-	T	-	VARCHAR	16	-	T
varbinary	1	8000	-	-	-	-	VARCHAR	-	-	T
varchar	1	8000	-	-	-	-	VARCHAR	-	-	N

Uwagi:

1. To odwzorowanie jest poprawne tylko w programie Microsoft SQL Server wersja 6.5.
2. To odwzorowanie jest poprawne tylko w programie Microsoft SQL Server wersja 7 i wersja 2000.
3. To odwzorowanie jest poprawne tylko w systemach Windows 2000.
4. To odwzorowanie jest poprawne tylko w programie Microsoft SQL Server 2000.

Źródła danych ODBC

Tabela 43. Domyślne proste odwzorowania typów danych dla źródeł ODBC (nie wszystkie kolumny pokazano)

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	-	-	T
SQL_BINARY	255	32672	-	-	-	-	VARCHAR	-	-	T
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N
SQL_CHAR	255	32672	-	-	-	-	VARCHAR	-	-	N
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	-	-	T
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	-	-	T
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

Źródła danych Oracle NET8

Tabela 44. Domyślne proste odwzorowania typów danych dla źródeł Oracle NET8 Server

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	T
CHAR	1	254	0	0	-	\0	CHAR	0	0	N
CHAR	255	2000	0	0	-	\0	VARCHAR	0	0	N
CLOB	0	0	0	0	-	\0	CLOB	2147483647	0	N
DATE	0	0	0	0	-	\0	TIMESTAMP	0	0	N
FLOAT	1	126	0	0	-	\0	DOUBLE	0	0	N
LONG	0	0	0	0	-	\0	CLOB	2147483647	0	N
LONG RAW	0	0	0	0	-	\0	BLOB	2147483647	0	T
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
NUMBER	-	10	0	0	-	\0	DECIMAL	0	0	N
RAW	1	2000	0	0	-	\0	VARCHAR	0	0	T
ROWID	0	0	0	NULL	-	\0	CHAR	18	0	N
TIMESTAMP ¹	-	-	-	-	-	-	TIMESTAMP	10	-	-
VARCHAR2	1	4000	0	0	-	\0	VARCHAR	0	0	N

Uwagi:

1. To odwzorowanie jest poprawne tylko dla konfiguracji klienta i serwera Oracle 9i (lub nowszej wersji).

Źródła danych Sybase

Tabela 45. Domyślne proste odwzorowania typów danych dla źródeł Sybase CTLIB

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	-	-	T
binary	255	16384	-	-	-	-	VARCHAR	-	-	T
bit	-	-	-	-	-	-	SMALLINT	-	-	-
char	1	254	-	-	-	-	CHAR	-	-	N
char	255	16384	-	-	-	-	VARCHAR	-	-	N
char null (patrz varchar)										
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
nchar null (patrz nvarchar)										
numeric	1	31	0	31	-	-	DECIMAL	-	-	-
numeric	32	38	0	38	-	-	DOUBLE	-	-	-
numericn	1	31	0	31	-	-	DECIMAL	-	-	-
numericn	32	38	0	38	-	-	DOUBLE	-	-	-
nvarchar	1	16384	-	-	-	-	VARCHAR	-	-	N
real	-	-	-	-	-	-	REAL	-	-	-

Tabela 45. Domyślne proste odwzorowania typów danych dla źródeł Sybase CTLIB (kontynuacja)

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
smalldatetime	-	-	-	-	-	-	TIMESTAMP	-	-	-
smallint	-	-	-	-	-	-	SMALLINT	-	-	-
smallmoney	-	-	-	-	-	-	DECIMAL	10	4	-
sysname	1	254	-	-	-	-	CHAR	-	-	N
text	-	-	-	-	-	-	CLOB	-	-	-
timestamp	-	-	-	-	-	-	VARCHAR	8	-	T
tinyint	-	-	-	-	-	-	SMALLINT	-	-	-
unichar ¹	1	254	-	-	-	-	CHAR	-	-	N
unichar ¹	255	16384	-	-	-	-	VARCHAR	-	-	N
unichar null (patrz univarchar)										
univarchar ¹	1	16384	-	-	-	-	VARCHAR	-	-	N
varbinary	1	16384	-	-	-	-	VARCHAR	-	-	T
varchar	1	16384	-	-	-	-	VARCHAR	-	-	N

Uwagi:

1. Poprawne dla stowarzyszonych baz danych z kodowaniem innym niż Unicode.

Źródła danych Teradata

Tabela 46. Domyślne proste odwzorowania typów danych dla źródeł Teradata (nie wszystkie kolumny pokazano)

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	-	-	T
BYTE	255	32672	-	-	-	-	VARCHAR	-	-	T
BYTE	32673	64000	-	-	-	-	BLOB	-	-	-
BYTEINT	-	-	-	-	-	-	SMALLINT	-	-	-

Tabela 46. Domyślne proste odwzorowania typów danych dla źródeł Teradata (nie wszystkie kolumny pokazano) (kontynuacja)

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	-	-	-	-	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	-	-	T
VARBYTE	32763	64000	-	-	-	-	BLOB	-	-	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
VARCHAR	32673	64000	-	-	-	-	CLOB	-	-	-
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	-	-
VARGRAPHIC	16337	32000	-	-	-	-	DBCLOB	-	-	-

Pojęcia pokrewne:

- “Proste i zwrotne odwzorowania typów danych” na stronie 50

Informacje pokrewne:

- “Zmiany długich typów danych na typy varchar” na stronie 57
- “Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie NET8” na stronie 297
- “Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie Sybase” na stronie 298

- “Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie ODBC” na stronie 299
- “Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie Microsoft SQL Server” na stronie 300

Rozdział 28. Domyślne zwrotne odwzorowania typów

Istnieją dwa rodzaje odwzorowań typów między typami źródła danych a typami w stowarzyszonej bazie danych: odwzorowania proste i odwzorowania zwrotne. W *prostym odwzorowaniu typów* typy zdalne są odwzorowane na odpowiednie typy lokalne. Drugim typem odwzorowania jest *odwzorowanie zwrotne*, używane do tworzenia lub modyfikowania zdalnych tabel za pomocą przezroczystych instrukcji DDL.

W przypadku większości źródeł danych domyślne odwzorowania typów są zdefiniowane w opakowaniach. Domyślne odwzorowania typów dla źródeł danych z rodziny DB2 określone są w opakowaniu DRDA. Domyślne odwzorowania typów dla źródeł Informix zawiera opakowanie INFORMIX, itd.

Definicja zdalnej tabeli lub widoku w stowarzyszonej bazie danych DB2 obejmuje zwrotne odwzorowanie typów. Odwzorowanie jest określone w kierunku od *lokalnego* typu danych programu DB2 dla systemów Linux, UNIX lub Windows dla danej kolumny do odpowiedniego *zdalnego* typu danych. Na przykład zgodnie z ustawieniem domyślnym lokalny typ REAL odwzorowany jest na typ Informix SMALLFLOAT.

Serwery stowarzyszone DB2 dla systemów Linux, UNIX i Windows nie obsługują odwzorowań dla typów danych LONG VARCHAR, LONG VARGRAPHIC, DATALINK oraz typów zdefiniowanych przez użytkownika.

Podczas tworzenia tabeli zdalnej za pomocą instrukcji CREATE TABLE określone są lokalne typy danych, które mają być włączone do tworzonej tabeli. Te domyślne zwrotne odwzorowania typów przypiszą tym kolumnom odpowiednie typy zdalne. Dla przykładu rozważmy instrukcję CREATE TABLE definiującą tabelę Informix z kolumną C2. W instrukcji jako typ danych kolumny C2 określono BIGINT. Domyślne odwzorowanie zwrotne typu BIGINT zależy od tego, w której wersji systemu Informix tworzona jest tabela. Odwzorowanie kolumny C2 w tabeli Informix będzie wskazywać typ DECIMAL w bazie danych Informix wersja 8 i typ INT8 w Informix wersja 9.

Można zmienić domyślne zwrotne odwzorowanie typów lub utworzyć nowe odwzorowanie, korzystając z instrukcji CREATE TYPE MAPPING.

W poniższych tabelach przedstawiono domyślne zwrotne odwzorowania między lokalnymi typami w bazach DB2 dla systemów Linux, UNIX i Windows a zdalnymi typami w źródłach danych.

Odwzorowania te obowiązują we wszystkich obsługiwanych wersjach programów, chyba że zaznaczono inaczej.

Źródła danych DB2 for z/OS and OS/390

Tabela 47. Domyślne zwrotne odwzorowania typów danych dla źródeł DB2 for z/OS and OS/390 (nie wszystkie kolumny pokazano)

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	-	-	-	-	T	-	CHAR	-	-	T
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
I DOUBLE	-	8	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	8	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
I REAL	-	4	-	-	-	-	REAL	-	-	-
I SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	T	-	VARCHAR	-	-	T
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	N

Źródła danych DB2 dla iSeries

Tabela 48. Domyślne zwrotne odwzorowania typów danych dla źródeł DB2 dla iSeries (nie wszystkie kolumny pokazano)

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	-	-	-	-	T	-	CHARACTER	-	-	T
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	-	-	-	-	T	-	VARCHAR	-	-	T
VARGRAPHIC	-	-	-	-	-	-	VARG	-	-	N

Źródła danych DB2 dla VM i VSE

Tabela 49. Domyślne zwrotne odwzorowania typów danych dla źródeł DB2 dla VM i VSE (nie wszystkie kolumny pokazano)

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	-	-	-	-	T	-	CHAR	-	-	T
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	-	-	-	-	T	-	VARCHAR	-	-	T
VARGRAPH	-	-	-	-	-	-	VARGRAPH	-	-	N

Źródła danych DB2 dla systemów Linux, UNIX i Windows

Tabela 50. Domyślne zwrotne odwzorowania typów danych dla źródeł DB2 dla systemów Linux, UNIX i Windows (nie wszystkie kolumny pokazano)

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	-	-	-	-	T	-	CHAR	-	-	T
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	-	-	-	-	T	-	VARCHAR	-	-	T
VARGRAPH	-	-	-	-	-	-	VARGRAPHIC	-	-	N
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	-

Źródła danych Informix

Tabela 51. Domyślne zwrotne odwzorowania typów danych dla źródeł Informix

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	-	-	-	-	T	-	BYTE	-	-	-
CLOB	1	2147483647	-	-	-	-	TEXT	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	SMALLFLOAT	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	DATETIME	6	10	-
TIMESTAMP	-	10	-	-	-	-	DATETIME	0	15	-
VARCHAR	1	254	-	-	N	-	VARCHAR	-	-	-
VARCHAR	255	32672	-	-	N	-	TEXT	-	-	-
VARCHAR	-	-	-	-	T	-	BYTE	-	-	-
VARCHAR ²	255	2048	-	-	N	-	LVARCHAR	-	-	-
VARCHAR ²	2049	32672	-	-	N	-	TEXT	-	-	-

Uwagi:

- To odwzorowanie typów jest poprawne tylko w przypadku serwerów Informix wersja 8 (lub starszych).
- To odwzorowanie typów jest poprawne tylko w przypadku serwerów Informix wersja 9.

W przypadku typu Informix DATETIME serwer stowarzyszony DB2 dla UNIX i Windows używa kwalifikatora wysokopoziomowego Informix jako parametru REMOTE_LENGTH, a kwalifikatora niskopoziomowego Informix jako parametru REMOTE_SCALE.

Kwalifikatory Informix są stałymi "TU_" zdefiniowanymi w pliku `datetime.h` za pomocą pakietu SDK klienta Informix. Stałe te, to:

0 = YEAR	8 = MINUTE	13 = FRACTION(3)
2 = MONTH	10 = SECOND	14 = FRACTION(4)
4 = DAY	11 = FRACTION(1)	15 = FRACTION(5)

Tabela 51. Domyślne zwrotne odwzorowania typów danych dla źródeł Informix (kontynuacja)

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
6 = HOUR										
		12 = FRACTION(2)								

Źródła danych Microsoft SQL Server

Tabela 52. Domyślne zwrotne odwzorowania typów danych dla źródeł Microsoft SQL Server (nie wszystkie kolumny pokazano)

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	-	-	-	-	-	-	bigint	-	-	-
BLOB	-	-	-	-	-	-	image	-	-	-
CHARACTER	-	-	-	-	T	-	binary	-	-	-
CHARACTER	-	-	-	-	N	-	char	-	-	-
CLOB	-	-	-	-	-	-	text	-	-	-
DATE	-	4	-	-	-	-	datetime	-	-	-
DECIMAL	-	-	-	-	-	-	decimal	-	-	-
DOUBLE	-	8	-	-	-	-	float	-	-	-
INTEGER	-	-	-	-	-	-	int	-	-	-
SMALLINT	-	-	-	-	-	-	smallint	-	-	-
REAL	-	4	-	-	-	-	real	-	-	-
TIME	-	3	-	-	-	-	datetime	-	-	-
TIMESTAMP	-	10	-	-	-	-	datetime	-	-	-
VARCHAR	1	8000	-	-	N	-	varchar	-	-	-
VARCHAR	8001	32672	-	-	N	-	text	-	-	-
VARCHAR	1	8000	-	-	T	-	varbinary	-	-	-

Tabela 52. Domyślne zwrotne odwzorowania typów danych dla źródeł Microsoft SQL Server (nie wszystkie kolumny pokazano) (kontynuacja)

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
VARCHAR	8001	32672	-	-	T	-	image	-	-	-

Uwagi:

1. To odwzorowanie jest poprawne tylko w programie Microsoft SQL Server 2000.

Źródła danych Oracle NET8

Tabela 53. Domyślne zwrotne odwzorowania typów danych dla źródeł Oracle NET8

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	T	\0	BLOB	0	0	T
CHARACTER	1	254	0	0	N	\0	CHAR	0	0	N
CHARACTER	1	254	0	0	T	\0	RAW	0	0	T
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
FLOAT	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

Tabela 53. Domyślne zwrotne odwzorowania typów danych dla źródeł Oracle NET8 (kontynuacja)

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
VARCHAR	1	2000	0	0	T	\0	RAW	0	0	T

Uwaga: Typ danych BIGINT, dostępny w programie DB2 Universal Database dla systemów Linux, UNIX i Windows, nie może być używany w przezroczystych instrukcjach DDL. Przy tworzeniu zdalnej tabeli Oracle nie można użyć typu danych BIGINT w instrukcji CREATE TABLE.

Źródła danych Sybase

Tabela 54. Domyślne zwrotne odwzorowania typów danych dla źródeł Sybase CTLIB

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	-	-	-	-	T	-	binary	-	-	-
CLOB	-	-	-	-	-	-	text	-	-	-
DATE	-	-	-	-	-	-	datetime	-	-	-
DECIMAL	-	-	-	-	-	-	decimal	-	-	-
DOUBLE	-	-	-	-	-	-	float	-	-	-
GRAPHIC	-	-	-	-	-	-	unichar	-	-	-
VARGRAPHIC	-	-	-	-	-	-	univarchar	-	-	-
INTEGER	-	-	-	-	-	-	integer	-	-	-
REAL	-	-	-	-	-	-	real	-	-	-
SMALLINT	-	-	-	-	-	-	smallint	-	-	-
TIME	-	-	-	-	-	-	datetime	-	-	-

Tabela 54. Domyślne zwrotne odwzorowania typów danych dla źródeł Sybase CTLIB (kontynuacja)

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
TIMESTAMP	-	-	-	-	-	-	datetime	-	-	-
VARCHAR ¹	1	255	-	-	N	-	varchar	-	-	-
VARCHAR ¹	256	32672	-	-	N	-	text	-	-	-
VARCHAR ²	1	16384	-	-	N	-	varchar	-	-	-
VARCHAR ²	16385	32672	-	-	N	-	text	-	-	-
VARCHAR ¹	1	255	-	-	T	-	varbinary	-	-	-
VARCHAR ¹	256	32672	-	-	T	-	image	-	-	-
VARCHAR ²	1	16384	-	-	T	-	varbinary	-	-	-
VARCHAR ²	16385	32672	-	-	T	-	image	-	-	-

Uwagi:

1. To odwzorowanie jest poprawne tylko dla źródeł CTLIB z serwerem Sybase w wersji 12.0 (lub starszej).
2. To odwzorowanie jest poprawne tylko dla źródeł CTLIB z serwerem Sybase w wersji 12.5 (lub nowszej).

Źródła danych Teradata

Tabela 55. Domyślne zwrotne odwzorowania typów danych dla źródeł Teradata (nie wszystkie kolumny pokazano)

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	-	-	-	-	T	-	BYTE	-	-	-
CLOB ²	1	64000	-	-	-	-	VARCHAR	-	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-

| Tabela 55. Domyślne zwrotne odwzorowania typów danych dla źródeł Teradata (nie wszystkie kolumny pokazano) (kontynuacja)

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
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	-	-	-	-	T	-	VARBYTE	-	-	-
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	-

| **Uwagi:**

- | 1. Typ danych Teradata VARBYTE może mieć tylko ściśle określoną długość (od 1 do 64000), bez względu na rzeczywistą długość danych typu DB2 BLOB.
- | 2. Typ danych Teradata VARCHAR może mieć tylko ściśle określoną długość (od 1 do 64000), bez względu na rzeczywistą długość danych typu DB2 CLOB.
- | 3. Typ danych Teradata VARGRAPHIC może mieć tylko ściśle określoną długość (od 1 do 32000), bez względu na rzeczywistą długość danych typu DB2 DBCLOB.

| **Pojęcia pokrewne:**

- “Proste i zwrotne odwzorowania typów danych” na stronie 50

Rozdział 29. Domyślne odwzorowania typów danych dla bazy danych Unicode

Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie NET8

Poniższa tabela przedstawia listę domyślnych prostych odwzorowań typów danych dla opakowania NET8, gdy stowarzyszona baza danych jest bazą danych Unicode.

Tabela 56. Domyślne proste odwzorowania typów danych dla opakowania NET8 i bazy danych Unicode

UTF-8		Oracle
Typ danych	Typ danych	Długość
CHAR	CHAR	od 1 do 254 bajtów
VARCHAR	CHAR	od 255 do 2000 bajtów
	VARCHAR2	od 1 do 4000 bajtów
DBCLOB	NCLOB	
GRAPHIC	NCHAR	od 1 do 127 znaków
VARGRAPHIC	NCHAR	od 128 do 1000 znaków
	NVARCHAR2	od 1 do 2000 znaków

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie NET8

Poniższa tabela przedstawia listę domyślnych zwrotnych odwzorowań typów danych dla opakowania NET8, gdy stowarzyszona baza danych jest bazą danych Unicode.

Tabela 57. Domyślne zwrotne odwzorowania typów danych dla opakowania NET8 i bazy danych Unicode

UTF-8		Oracle
Typ danych	Długość	Typ danych
CHAR	od 1 do 254 bajtów	CHAR
VARCHAR	od 1 do 4000 bajtów	VARCHAR2
CLOB	od 1 do 2 147 483 647 bajtów	CLOB
GRAPHIC	od 1 do 127 znaków	NCHAR
VARGRAPHIC	od 1 do 2000 znaków	NVARCHAR2
DBCLOB	od 1 do 1 073 741 823 znaków	NCLOB

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie Sybase

Poniższa tabela przedstawia listę domyślnych prostych odwzorowań typów danych dla opakowania CTLIB, gdy stowarzyszona baza danych jest bazą danych Unicode.

Tabela 58. Domyślne proste odwzorowania typów danych dla opakowania Sybase CTLIB i bazy danych Unicode

UTF-8	Sybase	
Typ danych	Typ danych	Długość
CHAR	char	od 1 do 254 bajtów
	nchar	od 1 do 127 znaków
VARCHAR	char	od 255 do 32672 bajtów
	varchar	od 1 do 32672 bajtów
	nchar	od 128 do 16336 znaków
	nvarchar	od 1 do 16336 znaków
CLOB	text	
GRAPHIC	unichar	od 1 do 127 znaków
VARGRAPHIC	unichar	od 128 do 16336 znaków
	univarchar	od 1 do 16336 znaków

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie Sybase

Poniższa tabela przedstawia listę domyślnych zwrotnych odwzorowań typów danych dla opakowania CTLIB, gdy stowarzyszona baza danych jest bazą danych Unicode.

Tabela 59. Domyślne zwrotne odwzorowania typów danych dla opakowania Sybase CTLIB i bazy danych Unicode

UTF-8	Sybase	
Typ danych	Długość	Typ danych
CHAR	od 1 do 254 bajtów	char
VARCHAR	od 1 do 32672 bajtów	varchar
CLOB	od 1 do 2 147 483 647 bajtów	text
GRAPHIC	od 1 do 127 znaków	unichar
VARGRAPHIC	od 1 do 16336 znaków	univarchar

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie ODBC

Poniższa tabela przedstawia listę domyślnych prostych odwzorowań typów danych dla opakowania ODBC, gdy stowarzyszona baza danych jest bazą danych Unicode.

Tabela 60. Domyślne proste odwzorowania typów danych dla opakowania ODBC

UTF-8	ODBC	
Typ danych	Typ danych	Długość
CHAR	SQL_CHAR	od 1 do 254 bajtów
VARCHAR	SQL_CHAR	od 255 do 32672 bajtów
	SQL_VARCHAR	od 1 do 32672 bajtów
CLOB	SQL_LONGVARCHAR	-
GRAPHIC	SQL_WCHAR	od 1 do 127 znaków
VARGRAPHIC	SQL_WVARCHAR	od 128 do 16336 znaków
	SQL_WVARCHAR	od 1 do 16336 znaków
DBCLOB	SQL_WLONGVARCHAR	-

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie ODBC

Poniższa tabela przedstawia listę domyślnych zwrotnych odwzorowań typów danych dla opakowania ODBC, gdy stowarzyszona baza danych jest bazą danych Unicode.

Tabela 61. Domyślne zwrotne odwzorowania typów danych dla opakowania ODBC

UTF-8	ODBC	
Typ danych	Długość	Typ danych
CHAR	od 1 do 254 bajtów	SQL_CHAR
VARCHAR	od 1 do 32672 bajtów	SQL_VARCHAR
CLOB	od 1 do 2 147 483 647 bajtów	SQL_LONGVARCHAR
GRAPHIC	od 1 do 127 znaków	SQL_WCHAR
VARGRAPHIC	od 1 do 16336 znaków	SQL_WVARCHAR
DBCLOB	od 1 do 1 073 741 823 znaków	SQL_WLONGVARCHAR

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Domyślne proste odwzorowania typów danych dla bazy danych Unicode - opakowanie Microsoft SQL Server

Poniższa tabela przedstawia listę domyślnych prostych odwzorowań typów danych dla opakowania Microsoft SQL Server, gdy stowarzyszona baza danych jest bazą danych Unicode.

Tabela 62. Domyślne proste odwzorowania typów danych dla opakowania Microsoft SQL Server

UTF-8	Microsoft SQL Server	
Typ danych	Typ danych	Długość
CHAR	CHAR	od 1 do 254 bajtów
VARCHAR	CHAR	od 255 do 8000 bajtów
	VARCHAR	od 1 do 8000 bajtów
CLOB	TEXT	-
GRAPHIC	NCHAR	od 1 do 127 znaków
VARGRAPHIC	NCHAR	od 128 do 16336 znaków
	NVARCHAR	od 1 do 16336 znaków
DBCLOB	NTEXT	-

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Domyślne zwrotne odwzorowania typów danych dla bazy danych Unicode - opakowanie Microsoft SQL Server

Poniższa tabela przedstawia listę domyślnych zwrotnych odwzorowań typów danych dla opakowania Microsoft SQL Server, gdy stowarzyszona baza danych jest bazą danych Unicode.

Tabela 63. Domyślne zwrotne odwzorowania typów danych dla opakowania Microsoft SQL Server

UTF-8	Microsoft SQL Server	
Typ danych	Długość	Typ danych
CHAR	od 1 do 254 bajtów	CHAR
VARCHAR	od 1 do 32672 bajtów	VARCHAR
CLOB	od 1 do 2 147 483 647 bajtów	TEXT
GRAPHIC	od 1 do 127 znaków	NCHAR
VARGRAPHIC	od 1 do 16336 znaków	NVARCHAR
DBCLOB	od 1 do 1 073 741 823 znaków	NTEXT

Pojęcia pokrewne:

- “Obsługa Unicode w systemach stowarzyszonych” na stronie 125

Rozdział 30. Typy danych obsługiwane dla nierelacyjnych źródeł danych

Dla większości nierelacyjnych źródeł danych, podczas tworzenia pseudonimów w celu uzyskania dostępu do danych z tych źródeł, konieczne jest określenie informacji o kolumnach, w tym typu danych.

Niektóre nierelacyjne opakowania tworzą wszystkie kolumny wymagane do uzyskania dostępu do źródła danych. Kolumny te nazywa się *kolumnami ustalonymi*. Inne opakowania umożliwiają samodzielne określenie typów danych dla niektórych lub wszystkich kolumn w instrukcji CREATE NICKNAME.

W poniższych sekcjach przedstawiono listę opakowań, dla których można określić typy danych, oraz typy danych obsługiwane przez te opakowania.

Typy danych obsługiwane przez opakowanie BioRS

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie BioRS.

Tabela 64. Typy danych BioRS, które można odwzorować na typy danych DB2

Typ danych BioRS	Typ danych DB2
AUTHOR	CHARACTER, CLOB, VARCHAR
DATE	CHARACTER, CLOB, VARCHAR
NUMBER	CHARACTER, CLOB, VARCHAR
REFERENCE	CHARACTER, CLOB, VARCHAR
TEXT	CHARACTER, CLOB, VARCHAR

Maksymalna dozwolona długość danych CLOB wynosi 5 megabajtów.

Typy danych obsługiwane przez opakowanie BLAST

Niektóre typy danych są automatycznie przypisywane kolumnom ustalonym tworzonym przez opakowanie BLAST.

Do pół wiersza definicji można przypisać typ danych podczas tworzenia pseudonimu. Jeśli typ danych w kolumnie wiersza definicji nie będzie zgodny z typem danych kolumny lokalnej, wystąpi błąd. Jeśli na przykład zdefiniuje się kolumnę wiersza definicji typu INTEGER, a kolumna ta będzie zawierać wartości inne niż liczbowe, zostanie zwrócony błąd.

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie BLAST.

Tabela 65. Typy danych BLAST, które można odwzorować na typy danych DB2

Typy danych BLAST	Typ danych DB2
wiersz definicji	CLOB
	Maksymalna dozwolona długość danych CLOB wynosi 5 megabajtów.
wiersz definicji	DOUBLE

Tabela 65. Typy danych BLAST, które można odwzorować na typy danych DB2 (kontynuacja)

Typy danych BLAST	Typ danych DB2
wiersz definicji	FLOAT
wiersz definicji	INTEGER
wiersz definicji	VARCHAR

Typy danych obsługiwane przez opakowanie Documentum

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie Documentum.

Tabela 66. Typy danych Documentum, które można odwzorować na typy danych DB2

Typy danych Documentum	Typ danych DB2
DOUBLE	DOUBLE, FLOAT, INTEGER, SMALLINT
ID	CHARACTER (16)
INTEGER	DOUBLE, FLOAT, INTEGER, SMALLINT
STRING (maks. 255 znaków)	CHAR, VARCHAR
TIME	CHAR, DATE, TIMESTAMP, VARCHAR

Typy danych obsługiwane przez opakowanie Entrez

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie Entrez.

Tabela 67. Typy danych Entrez, które można odwzorować na typy danych DB2

Typy danych Entrez	Typ danych DB2
znak	CHARACTER
znak	CLOB
	Maksymalna dozwolona długość danych CLOB wynosi 5 megabajtów.
data	DATE
liczba	DECIMAL
liczba	DOUBLE
liczba całkowita	INTEGER
liczba	REAL
liczba całkowita	SMALLINT
time	TIMESTAMP
znak	VARCHAR

Typy danych obsługiwane przez opakowanie Excel

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie Excel.

Tabela 68. Typy danych Excel, które można odwzorować na typy danych DB2

Typy danych Excel	Typ danych DB2
data	DATE
liczba	DOUBLE
liczba	FLOAT (n) gdzie n oznacza ≥ 25 i ≤ 53
liczba całkowita	INTEGER
znak	VARCHAR

Typy danych obsługiwane przez opakowanie Extended Search

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie Extended Search.

Tabela 69. Typy danych Extended Search, które można odwzorować na typy danych DB2

Typy danych Extended Search	Typ danych DB2
Date	DATE
Double	DOUBLE
Integer	INTEGER
String	VARCHAR

Typy danych obsługiwane przez opakowanie HMMER

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie HMMER.

Tabela 70. Typy danych HMMER, które można odwzorować na typy danych DB2

Typy danych HMMER	Typ danych DB2
znak	CLOB
	Maksymalna dozwolona długość danych CLOB wynosi 5 megabajtów.
znak	DOUBLE
znak	FLOAT
znak	INTEGER
znak	VARCHAR

Typy danych obsługiwane przez opakowanie dla plików o strukturze tabeli

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie dla plików o strukturze tabeli.

Tabela 71. Typy danych plików o strukturze tabeli, które można odwzorować na typy danych DB2

Typy danych plików o strukturze tabeli	Typ danych DB2
znak	CHARACTER
znak	CLOB
	Maksymalna dozwolona długość danych CLOB wynosi 5 megabajtów.
liczba	DECIMAL
liczba	DOUBLE
liczba	FLOAT
liczba całkowita	INTEGER
liczba	REAL
liczba całkowita	SMALLINT
znak	VARCHAR

Typy danych obsługiwane przez opakowanie dla usług WWW

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie dla usług WWW. Opakowanie dla usług WWW używa typów danych XML.

Tabela 72. Typy danych XML, które można odwzorować na typy danych DB2, dla opakowania usług WWW

Typy danych XML	Typ danych DB2
znak	CHARACTER
znak	CHARACTER FOR BIT DATA
znak	CLOB
data	DATE
liczba	DECIMAL
liczba	DOUBLE
liczba	FLOAT
liczba całkowita	INTEGER
liczba	REAL
liczba całkowita	SMALLINT
znak	VARCHAR
znak	VARCHAR FOR BIT DATA

Typy danych obsługiwane przez opakowanie WebSphere Business Integration

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie WebSphere Business Integration. Opakowanie WebSphere Business Integration używa typów danych XML.

Tabela 73. Typy danych XML, które można odwzorować na typy danych DB2, dla opakowania WebSphere Business Integration

Typy danych XML	Typ danych DB2
znak	CHARACTER
znak	CHARACTER FOR BIT DATA
znak	CLOB
data	DATE
liczba	DECIMAL
liczba	DOUBLE
liczba	FLOAT
liczba całkowita	INTEGER
liczba	REAL
liczba całkowita	SMALLINT
znak	VARCHAR
znak	VARCHAR FOR BIT DATA

Typy danych obsługiwane przez opakowanie XML

Poniższa tabela przedstawia listę typów danych programu DB2 obsługiwanych przez opakowanie XML.

Tabela 74. Typy danych XML, które można odwzorować na typy danych DB2, dla opakowania XML

Typy danych XML	Typ danych DB2
znak	CHARACTER
znak	CHARACTER FOR BIT DATA
znak	CLOB
	Maksymalna dozwolona długość danych CLOB wynosi 5 megabajtów.
data	DATE
liczba	DECIMAL
liczba	DOUBLE
liczba	FLOAT
liczba całkowita	INTEGER
liczba	REAL
liczba całkowita	SMALLINT
znak	VARCHAR
znak	VARCHAR FOR BIT DATA

Rozdział 31. Elementy monitora systemu stowarzyszonej bazy danych

System stowarzyszony jest to serwer, za pośrednictwem którego realizowany jest zdalny dostęp do wielu baz danych. Udostępnia on na potrzeby klientów rozmaite źródła danych, które mogą rezydować na różnych platformach pochodzących od firmy IBM lub innych dostawców, zarówno relacyjnych, jak i nierelacyjnych. Rola serwera polega na takim ujednoczeniu drogi dostępu do danych rozproszonych, aby środowisko heterogeniczne od strony użytkowników sprawiało wrażenie pojedynczej bazy danych.

Poniżej zestawiono listę źródeł dotyczących dostępu do danych przez aplikacje działające w ramach systemu stowarzyszonego DB2 w ujęciu ogólnym, jak i w kontekście danej aplikacji w danej instancji serwera stowarzyszonego. Są to:

- `datasource_name` - Data Source Name monitor element
- `disconnects` - Disconnects monitor element
- `insert_sql_stmts` - Inserts monitor element
- `update_sql_stmts` - Updates monitor element
- `delete_sql_stmts` - Deletes monitor element
- `create_nickname` - Create Nicknames monitor element
- `passthru` - Pass-Through monitor element
- `stored_procs` - Stored Procedures monitor element
- `remote_locks` - Remote Locks monitor element
- `sp_rows_selected` - Rows Returned by Stored Procedures monitor element
- `select_time` - Query Response Time monitor element
- `insert_time` - Insert Response Time monitor element
- `update_time` - Update Response Time monitor element
- `delete_time` - Delete Response Time monitor element
- `create_nickname_time` - Create Nickname Response Time monitor element
- `passthru_time` - Pass-Through Time monitor element
- `stored_proc_time` - Stored Procedure Time monitor element
- `remote_lock_time` - Remote Lock Time monitor element

Rozdział 32. Procedura zapisana w bazie SYSPROC.NNSTAT

Pobiera dostępne w danej chwili statystyki pseudonimów.

Autoryzacja

Brak.

Składnia

```
CALL SYSPROC.NNSTAT(  
    'server/null', 'schema/null', 'nickname/null', 'filepath/null',?,?)
```

Opis parametrów

Server Serwer, na którym serwer stowarzyszony zbiera statystyki pseudonimów. Jest to serwer rejestrowany przez użytkowników w celu zdefiniowania źródła danych w stowarzyszonej bazie danych. Jeśli określono jeden pseudonim, dla tego parametru można określić wartość NULL.

Schema

Podanie wartości NULL dla tego parametru spowoduje, że serwer stowarzyszony pobierze statystyki dla wszystkich pseudonimów w ramach danego serwera. Jeśli parametr Server ma wartość NULL, serwer stowarzyszony pobierze statystyki pseudonimu w tym schemacie. Jeśli parametry Schema i Nickname mają wartość NULL, ale serwer został określony, serwer stowarzyszony pobierze statystyki dla tego serwera.

Nickname

Nazwa pseudonimu. Jeśli określi się pseudonim, konieczne jest także określenie schematu.

Log_File_Path

Ścieżka i nazwa pliku protokołu. Serwer stowarzyszony utworzy plik protokołu na serwerze. Katalogi podane w ścieżce muszą istnieć. W systemie Windows w ścieżce do protokołu należy użyć dwóch ukośników odwrotnych. Na przykład: c:\\temp\\nnstat.log. Określenie dla tego parametru wartości NULL spowoduje, że serwer stowarzyszony nie utworzy protokołu.

Parametry wyjściowe

out_SQLCode

Błąd SQL, który wystąpił podczas pobierania statystyk.

out_Trace

Ślad.

Przykłady

```
CALL SYSPROC.NNSTAT(  
    'NULL', 'ADMIN', 'STAFF', '/home/iiuser/reportlogs/log1.txt',?,?)  
CALL SYSPROC.NNSTAT(  
    'DB2SERV', 'ADMIN', 'NULL', 'c:\\raporty\\log1.txt',?,?)  
CALL SYSPROC.NNSTAT(  
    'DB2SERV', 'NULL', 'NULL', 'NULL',?,?)
```

Pojęcia pokrewne:

- “Narzędzie do aktualizacji statystyk pseudonimu - przegląd” na stronie 191

Zadania pokrewne:

- “Pobieranie statystyk pseudonimów” na stronie 192

Informacje pokrewne:

- “Pobieranie statystyk pseudonimów z wiersza komend - przykłady” na stronie 194

Rozdział 33. Usuwanie skutków awarii w środowiskach o wysokiej dostępności a stowarzyszone źródła danych

W systemach usuwania skutków awarii w środowiskach o wysokiej dostępności (HADR - high availability disaster recovery) podstawowa baza danych przesyła protokoły do dodatkowej bazy danych. Jeśli podstawowa baza danych będzie niedostępna, można przełączyć się na dodatkową bazę danych.

Wymagania dla opakowań relacyjnych

W celu zaimplementowania systemu HADR w stowarzyszonej bazie danych muszą być spełnione następujące warunki:

- Nazwy partycji źródła danych w obydwu systemach muszą być identyczne. Jeśli na przykład system z podstawową bazą danych ma partycję o nazwie INF1, która odwołuje się do instancji programu Informix na hoście xyz, wówczas w systemie z dodatkową bazą danych partycja INF1 musi odwoływać się do tej samej instancji programu Informix na tym samym hoście.
- W przypadku źródeł danych Oracle wersja programu klienta w systemie z podstawową bazą danych musi być taka sama, jak wersja programu klienta w systemie z dodatkową bazą danych. W przypadku innych źródeł danych wersje programów klienta muszą być takie same.
- Poniższe zmienne środowiskowe w stowarzyszonym pliku konfiguracyjnym muszą być takie same w systemach z podstawową i dodatkową bazą danych:

Oracle: NLS_LANG

Informix: CLIENT_LOCALE, DB_LOCALE, DBNLS

Sybase: SYBASE_CHARSET

Teradata: TERADATA_CHARSET

Wymagania dla opakowań nierelacyjnych

Dla plików tekstowych o strukturze tabeli, opakowań XML i opakowań Excel ścieżki dostępu do plików zewnętrznych muszą być takie same w obydwu systemach.

Dla opakowań HMMER i BLAST każdy system musi mieć dostęp TCP/IP do odpowiedniego demona, o ile ten demon jest uruchomiony na innym komputerze niż program DB2 Information Integrator.

Dla opakowania Documentum klient programu Documentum musi być zainstalowany w systemie dodatkowym. Pliki konfiguracyjne programu klienta muszą kierować połączenia do tego samego procesu docbroker.

Dla programu WebSphere Business Integration:

- Jeśli menedżer WebSphere MQ i opakowanie nie znajdują się w tym samym systemie, konieczne jest zainstalowanie programu klienta WebSphere MQ w systemie dodatkowym. System dodatkowy musi mieć dostęp do tego samego menedżera WebSphere MQ co system podstawowy.
- Jeśli menedżer WebSphere MQ i opakowanie znajdują się w tym samym systemie, ale adapter znajduje się na innym komputerze, w systemie dodatkowym musi działać menedżer WebSphere MQ. W razie awarii systemu podstawowego konieczne jest zamknięcie adaptera i przekonfigurowanie go, aby wskazywał nowego menedżera WebSphere MQ. Nowy menedżer WebSphere MQ musi mieć zdefiniowane takie same

nazwy kolejek, jak w systemie podstawowym. Następnie należy restartować adapter w celu połączenia z dodatkowym menedżerem WebSphere MQ.

- Gdy menedżer WebSphere MQ, opakowanie i adapter znajdują się w tym samym systemie, konieczne jest utworzenie repliki tej konfiguracji na komputerze dodatkowym.

Pojęcia pokrewne:

- “High availability disaster recovery overview” w podręczniku *Data Recovery and High Availability Guide and Reference*

Rozdział 34. Informacje o serwerze bramy zapytań dla pułapek mechanizmu

Aby ułatwić pracownikom Serwisu IBM diagnozowanie awarii mechanizmu programu DB2 UDB, program DB2 UDB może zapisywać w pliku niektóre informacje na temat serwera bramy zapytań. Aby zapisać te informacje w pliku:

- Parametrowi konfiguracyjnemu menedżera bazy danych FEDERATED nadaj wartość YES.
- Utwórz co najmniej jeden obiekt serwera.
- Aby zapisać w tym samym pliku informacje o puli pamięci bramy zapytań, nadaj zmiennej rejestru DB2MEMDBG wartość ON.

Po wystąpieniu pułapki program protokołujący zapisze komunikat w pliku db2diag.log. W komunikacie będzie podana pełna ścieżka do pliku z informacjami o serwerze bramy zapytań.

Dokumentacja techniczna programu DB2 Information Integrator

Ta sekcja zawiera informacje na temat:

- dokumentacji PDF programu DB2 Information Integrator
- dokumentów Release Notes oraz Wymagania instalacyjne dla programu DB2 Information Integrator
- pakietów poprawek dokumentacji programu DB2 Information Integrator

Dokumentacja programu DB2 Information Integrator

Ten temat obejmuje informacje dotyczące dokumentacji dla programu DB2 Information Integrator. Zamieszczone w nim tabele zawierają oficjalny tytuł dokumentu, numer zamówienia oraz położenie każdego podręcznika w formacie PDF. Aby zamówić podręcznik wydrukowany, niezbędna jest znajomość oficjalnego tytułu podręcznika lub numeru zamówienia. Ten temat zawiera również informacje o tytułach, nazwach plików i położeniu takich dokumentów jak uwagi do wydania i wymagania instalacyjne.

Ten temat obejmuje następujące sekcje:

- Dostęp do dokumentacji programu DB2 Information Integrator
- Dokumentacja funkcji replikacji w systemie z/OS
- Dokumentacja funkcji publikowania zdarzeń programu DB2 Universal Database w systemie z/OS
- Dokumentacja funkcji publikowania zdarzeń dla IMS i VSAM w systemie z/OS
- Dokumentacja funkcji publikowania zdarzeń i replikacji w systemach Linux, UNIX i Windows
- Dokumentacja funkcji stowarzyszonej w systemie z/OS
- Dokumentacja funkcji stowarzyszonej w systemach Linux, UNIX i Windows
- Dokumentacja programu Enterprise Search w systemach Linux, UNIX i Windows
- Uwagi do wydania i wymagania instalacyjne
- Uwagi

Dostęp do dokumentacji programu DB2 Information Integrator

Wszystkie podręczniki programu DB2 Information Integrator oraz uwagi do wydania są dostępne w plikach PDF w serwisie WWW programu DB2 Information Integrator pod adresem www.ibm.com/software/data/integration/db2ii/support.html.

Aby uzyskać dostęp do najnowszej dokumentacji produktu DB2 Information Integrator, w serwisie WWW programu DB2 Information Integrator kliknij odsyłacz Product information (Informacja o produkcie), jak pokazano na rysunku Rys. 10 na stronie 316.

The screenshot shows the IBM DB2 Information Integrator website. The top navigation bar includes 'Home', 'Products & services', 'Support & downloads', and 'My account'. The breadcrumb trail is 'Software > DB2 Information Management > DB2 Information Integration >'. The main heading is 'DB2 Information Integrator'. A search bar is present with the text 'Search support for this product' and 'Enter search terms, phrase, error code or APAR number'. Below the search bar are options to 'Solve a problem (FAQs, APARs, Technotes)', 'Download (Fixes, Patches)', and 'Learn (Manual Papers, etc.)'. A 'Submit' button is highlighted with a blue circle. Below the search bar are links for 'Advanced search for this product' and 'Search all software support'. On the left, a sidebar contains a 'Support' section with a list of links: 'My support', 'Submit & track problems', 'How to buy software support', 'Help', 'Site tours', and 'Feedback'. The 'Learn' section is circled in blue, containing links for 'Product information' (highlighted in yellow) and 'eBooks'. The 'Self help' section includes 'Solve a problem' (with a link to 'Technotes'), 'Download' (with links to 'Get Statistics Utilities' and 'Version 8.1 Fix Packs'), and 'Learn' (with links to 'Product information' and 'eBooks'). The 'Problem submission' section includes a link to 'Submit & track problem:' and a link to 'How to buy support for your software'. The 'Other resources' section is also visible.

Rysunek 10. Dostęp do odsyłacza Product information (Informacja o produkcie) w serwisie WWW programu DB2 Information Integrator

Poprzez odsyłacz Product information (Informacja o produkcie) można uzyskać dostęp do najnowszej dokumentacji programu DB2 Information Integrator we wszystkich obsługiwanych językach:

- Dokumentacja produktu DB2 Information Integrator w plikach w formacie PDF
- Dokumentacja pakietu poprawek, w tym uwagi do wydania
- Instrukcje pobierania i instalowania programu Centrum informacyjne DB2 dla systemów operacyjnych Linux, UNIX i Windows
- Odsyłacze do elektronicznego Centrum informacyjnego DB2

Przewin listę, aby znaleźć dokumentację produktu dla używanej wersji programu DB2 Information Integrator.

W serwisie WWW produktu DB2 Information Integrator udostępniona jest również dokumentacja pomocnicza, dokumentacja techniczna IBM (Redbooks), raporty, zasoby, odsyłacze do grup użytkowników oraz najnowsze informacje na temat programu DB2 Information Integrator.

Podręczniki programu DB2 Information Integrator w formacie PDF można również przeglądać i drukować z dysku CD-ROM *Dokumentacja DB2 w formacie PDF*.

Aby przejrzeć lub wydrukować dokumentację w formacie PDF:

1. W katalogu głównym dysku CD *Dokumentacja DB2 w formacie PDF* otwórz plik `index.htm`.
2. Kliknij nazwę wybranego języka.
3. Kliknij odsyłacz do dokumentu, który chcesz przejrzeć.

Dokumentacja funkcji replikacji w systemie z/OS

Tabela 75. Dokumentacja programu DB2 Information Integrator dotycząca funkcji replikacji w systemie z/OS

Nazwa	Numer zamówienia	Położenie
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Introduction to Replication and Event Publishing</i>	GC18-7567	Serwis WWW produktu DB2 Information Integrator
<i>Migrating to SQL Replication</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> • Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> • Serwis WWW produktu DB2 Information Integrator
<i>Replication Installation and Customization Guide for z/OS</i>	SC18-9127	Serwis WWW produktu DB2 Information Integrator
<i>SQL Replication Guide and Reference</i>	SC27-1121	<ul style="list-style-type: none"> • Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> • Serwis WWW produktu DB2 Information Integrator
<i>Tuning for Replication and Event Publishing Performance</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Tuning for SQL Replication Performance</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator

Tabela 75. Dokumentacja programu DB2 Information Integrator dotycząca funkcji replikacji w systemie z/OS (kontynuacja)

Nazwa	Numer zamówienia	Położenie
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	Nie dotyczy	<ul style="list-style-type: none"> W Centrum informacyjnym DB2: Przegląd produktów > Information Integration > Przegląd programu DB2 Information Integrator > Problemy, obejścia i aktualizacje dokumentacji Wyrzutnia instalacji programu DB2 Information Integrator Serwis WWW produktu DB2 Information Integrator Dysk CD programu <i>DB2 Information Integrator</i>

Dokumentacja funkcji publikowania zdarzeń programu DB2 Universal Database w systemie z/OS

Tabela 76. Dokumentacja programu DB2 Information Integrator dotycząca funkcji publikowania zdarzeń programu DB2 Universal Database w systemie z/OS

Nazwa	Numer zamówienia	Położenie
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Introduction to Replication and Event Publishing</i>	GC18-7567	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>Tuning for Replication and Event Publishing Performance</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	Nie dotyczy	<ul style="list-style-type: none"> W Centrum informacyjnym DB2: Przegląd produktów > Information Integration > Przegląd programu DB2 Information Integrator > Problemy, obejścia i aktualizacje dokumentacji Wyrzutnia instalacji programu DB2 Information Integrator Serwis WWW produktu DB2 Information Integrator Dysk CD programu <i>DB2 Information Integrator</i>

Dokumentacja funkcji publikowania zdarzeń dla IMS i VSAM w systemie z/OS

Tabela 77. Dokumentacja programu DB2 Information Integrator dotycząca funkcji publikowania zdarzeń dla IMS i VSAM w systemie z/OS

Nazwa	Numer zamówienia	Położenie
<i>Client Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9160	Serwis WWW produktu DB2 Information Integrator
<i>Data Mapper Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9163	Serwis WWW produktu DB2 Information Integrator
<i>Getting Started with Event Publisher for z/OS</i>	GC18-9186	Serwis WWW produktu DB2 Information Integrator
<i>Installation Guide for Classic Federation and Event Publisher for z/OS</i>	GC18-9301	Serwis WWW produktu DB2 Information Integrator
<i>Operations Guide for Event Publisher for z/OS</i>	SC18-9157	Serwis WWW produktu DB2 Information Integrator
<i>Planning Guide for Event Publisher for z/OS</i>	SC18-9158	Serwis WWW produktu DB2 Information Integrator
<i>Reference for Classic Federation and Event Publisher for z/OS</i>	SC18-9156	Serwis WWW produktu DB2 Information Integrator
<i>System Messages for Classic Federation and Event Publisher for z/OS</i>	SC18-9162	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for IMS for z/OS</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for VSAM for z/OS</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator

Dokumentacja funkcji publikowania zdarzeń i replikacji w systemach Linux, UNIX i Windows

Tabela 78. Dokumentacja programu DB2 Information Integrator dotycząca funkcji publikowania zdarzeń i replikacji w systemach Linux, UNIX i Windows

Nazwa	Numer zamówienia	Położenie
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Podręcznik instalowania dla systemów Linux, UNIX i Windows</i>	GC85-0105	<ul style="list-style-type: none"> • Dysk CD Dokumentacja DB2 w formacie PDF • Serwis WWW produktu DB2 Information Integrator
<i>Introduction to Replication and Event Publishing</i>	GC18-7567	<ul style="list-style-type: none"> • Dysk CD Dokumentacja DB2 w formacie PDF • Serwis WWW produktu DB2 Information Integrator
<i>Migrating to SQL Replication</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator

Tabela 78. Dokumentacja programu DB2 Information Integrator dotycząca funkcji publikowania zdarzeń i replikacji w systemach Linux, UNIX i Windows (kontynuacja)

Nazwa	Numer zamówienia	Położenie
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>SQL Replication Guide and Reference</i>	SC27-1121	Serwis WWW produktu DB2 Information Integrator
<i>Tuning for Replication and Event Publishing Performance</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Tuning for SQL Replication Performance</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	Nie dotyczy	<ul style="list-style-type: none"> W Centrum informacyjnym DB2: Przegląd produktów > Information Integration > Przegląd programu DB2 Information Integrator > Problemy, obejścia i aktualizacje dokumentacji Wyrzutnia instalacji programu DB2 Information Integrator Serwis WWW produktu DB2 Information Integrator Dysk CD programu <i>DB2 Information Integrator</i>

Dokumentacja funkcji stowarzyszonej w systemie z/OS

Tabela 79. Dokumentacja programu DB2 Information Integrator dotycząca funkcji stowarzyszonej w systemie z/OS

Nazwa	Numer zamówienia	Położenie
<i>Client Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9160	Serwis WWW produktu DB2 Information Integrator
<i>Data Mapper Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9163	Serwis WWW produktu DB2 Information Integrator
<i>Getting Started with Classic Federation for z/OS</i>	GC18-9155	Serwis WWW produktu DB2 Information Integrator
<i>Installation Guide for Classic Federation and Event Publisher for z/OS</i>	GC18-9301	Serwis WWW produktu DB2 Information Integrator
<i>Reference for Classic Federation and Event Publisher for z/OS</i>	SC18-9156	Serwis WWW produktu DB2 Information Integrator
<i>System Messages for Classic Federation and Event Publisher for z/OS</i>	SC18-9162	Serwis WWW produktu DB2 Information Integrator
<i>Transaction Services Guide for Classic Federation for z/OS</i>	SC18-9161	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for IBM DB2 Information Integrator Classic Federation for z/OS</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator

Dokumentacja funkcji stowarzyszonej w systemach Linux, UNIX i Windows

Tabela 80. Dokumentacja programu DB2 Information Integrator dotycząca funkcji stowarzyszonej w systemach Linux, UNIX, and Windows

Nazwa	Numer zamówienia	Położenie
<i>Application Developer's Guide</i>	SC18-7359	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>C++ API Reference for Developing Wrappers</i>	SC18-9172	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>Data Source Configuration Guide</i>	Nie dotyczy	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>Podręcznik systemów stowarzyszonych</i>	SC85-0104	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>Guide to Configuring the Content Connector for VeniceBridge</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Podręcznik instalowania dla systemów Linux, UNIX i Windows</i>	GC85-0105	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>Java API Reference for Developing Wrappers</i>	SC18-9173	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>Migration Guide</i>	SC18-7360	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator
<i>Wrapper Developer's Guide</i>	SC18-9174	<ul style="list-style-type: none"> Dysk CD <i>Dokumentacja DB2 w formacie PDF</i> Serwis WWW produktu DB2 Information Integrator

Tabela 80. Dokumentacja programu DB2 Information Integrator dotycząca funkcji stowarzyszonej w systemach Linux, UNIX, and Windows (kontynuacja)

Nazwa	Numer zamówienia	Położenie
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	Nie dotyczy	<ul style="list-style-type: none"> W Centrum informacyjnym DB2: Przegląd produktów > Information Integration > Przegląd programu DB2 Information Integrator > Problemy, obejścia i aktualizacje dokumentacji Wyrzutnia instalacji programu DB2 Information Integrator Serwis WWW produktu DB2 Information Integrator Dysk CD programu <i>DB2 Information Integrator</i>

Dokumentacja funkcji Enterprise Search w systemach Linux, UNIX i Windows

Tabela 81. Dokumentacja programu DB2 Information Integrator dotycząca funkcji Enterprise Search w systemach Linux, UNIX i Windows

Nazwa	Numer zamówienia	Położenie
<i>Administering Enterprise Search</i>	SC18-9283	Serwis WWW produktu DB2 Information Integrator
<i>Installation Guide for Enterprise Search</i>	GC18-9282	Serwis WWW produktu DB2 Information Integrator
<i>Programming Guide and API Reference for Enterprise Search</i>	SC18-9284	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for Enterprise Search</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator

Uwagi do wydania oraz wymagania instalacyjne

Uwagi do wydania (Release notes) dostarczają informacji dotyczących konkretnego wydania i poziomu pakietu poprawek produktu i obejmują najnowsze poprawki do dokumentacji dla każdego wydania.

Wymagania instalacyjne dostarczają informacji dotyczących konkretnego wydania produktu.

Tabela 82. Uwagi do wydania i wymagania instalacyjne programu DB2 Information Integrator

Nazwa	Nazwa pliku	Położenie
<i>Wymagania instalacyjne dla programu IBM DB2 Information Integrator Event Publishing Edition, Replication Edition, Standard Edition, Advanced Edition, Advanced Edition Unlimited, Developer Edition i Replication dla systemu z/OS</i>	Prereqs	<ul style="list-style-type: none"> Dysk CD programu <i>DB2 Information Integrator</i> Wyrzutnia instalacji programu DB2 Information Integrator

Tabela 82. Uwagi do wydania i wymagania instalacyjne programu DB2 Information Integrator (kontynuacja)

Nazwa	Nazwa pliku	Położenie
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	ReleaseNotes	<ul style="list-style-type: none"> W Centrum informacyjnym DB2: Przegląd produktów > Information Integration > Przegląd programu DB2 Information Integrator > Problemy, obejścia i aktualizacje dokumentacji Wyrzutnia instalacji programu DB2 Information Integrator Serwis WWW produktu DB2 Information Integrator Dysk CD programu <i>DB2 Information Integrator</i>
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for IMS for z/OS</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for VSAM for z/OS</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for IBM DB2 Information Integrator Classic Federation for z/OS</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator
<i>Release Notes for Enterprise Search</i>	Nie dotyczy	Serwis WWW produktu DB2 Information Integrator

Aby przejrzeć znajdujące się na dysku CD produktu wymagania instalacyjne i uwagi do wydania:

- W systemach operacyjnych Windows wprowadź:
`x:\doc\%L`
x oznacza literę napędu CD w systemie Windows, a *%L* oznacza ustawienia narodowe dokumentacji, którą chcesz przejrzeć, na przykład pl_PL.
- W systemach operacyjnych UNIX wprowadź:
`/cdrom/doc/%L/`
cdrom oznacza punkt podłączenia napędu CD w systemie UNIX, a *%L* oznacza ustawienia narodowe dokumentacji, którą chcesz przejrzeć, na przykład pl_PL.

Uwagi do wydania oraz wymagania instalacyjne

Uwagi do wydania i wymagania instalacyjne dostarczają informacji odnoszących się do konkretnego wydania i wersji pakietu poprawek produktu. Zawierają również zestawienia uaktualnień dokumentacji włączanych do każdego wydania i pakietu poprawek.

Uwagi do wydania i wymagania instalacyjne są dostępne w formacie tekstowym i formacie HTML na dysku CD-ROM produktu:

- W systemach operacyjnych Windows: `x:\doc\%L`
- W systemach operacyjnych UNIX: `/cdrom/doc/%L/`

gdzie:

- x* reprezentuje literę napędu CD-ROM w systemie Windows,
- cdrom* oznacza punkt podłączenia napędu CD-ROM w systemie UNIX,

- *%L* określa ustawienie narodowe dokumentacji, która ma zostać użyta; na przykład en_US.

Poniższa tabela zawiera informacje na temat położenia Uwag do wydania oraz wymagań instalacyjnych.

Tabela 83. Uwagi do wydania i wymagania instalacyjne.

Nazwa	Nazwa pliku	Położenie
<i>IBM DB2 Information Integrator Release Notes</i>	ReleaseNotes	<ul style="list-style-type: none"> • Uwagi do wydania w Centrum informacyjnym DB2 • W sieci WWW należy przejść na stronę: http://www.ibm.com/software/data/integration/db2ii/support.html
<i>IBM DB2 Information Integrator - Wymagania instalacyjne</i>	Prereqs	<ul style="list-style-type: none"> • Dysk CD produktu • Wyrzutnia instalacji programu DB2 Information Integration

Pakiety poprawek do dokumentacji programu DB2 Information Integrator

Firma IBM może co pewien czas publikować pakiety poprawek i inne aktualizacje dokumentacji dla Centrum informacyjnego DB2. Centrum informacyjne DB2, znajdujące się pod adresem <http://publib.boulder.ibm.com/infocenter/db2help/>, zawiera zawsze najbardziej aktualne informacje. Jeśli Centrum informacyjne DB2 zostało zainstalowane lokalnie, to wszystkie aktualizacje trzeba instalować ręcznie, zanim będzie można je wyświetlać. Aktualizacje dokumentacji umożliwiają zaktualizowanie informacji zainstalowanych z dysku CD-ROM *Centrum informacyjne DB2* w chwili, gdy nowe informacje stają się dostępne.

Centrum informacyjne jest aktualizowane częściej niż pliki PDF i podręczniki drukowane. Aby mieć dostęp do najbardziej aktualnych informacji technicznych DB2, należy instalować dostępne aktualizacje dokumentacji lub korzystać z Centrum informacyjnego DB2 w serwisie www.ibm.com.

Ułatwienia dostępu

Ułatwienia dostępu pomagają użytkownikom niepełnosprawnym fizycznie, na przykład z upośledzeniem ruchowym lub wzrokowym, efektywnie korzystać z oprogramowania. Poniższa lista zawiera opis głównych ułatwień dostępu w produktach DB2[®], wersja 8:

- Ze wszystkich funkcji programu DB2 można korzystać za pośrednictwem klawiatury, bez konieczności użycia myszy. Więcej informacji na ten temat można znaleźć w sekcji “Wprowadzanie danych i nawigacja za pomocą klawiatury”.
- Interfejsy użytkownika programu DB2 umożliwiają dostosowanie wielkości i koloru czcionek. Więcej informacji na ten temat można znaleźć w sekcji “Przystępny ekran”.
- Produkty DB2 obsługują ułatwiające dostęp aplikacje korzystające z interfejsu Java[™] Accessibility API. Więcej informacji na ten temat można znaleźć w sekcji “Zgodność z rozwiązaniami technicznymi dla niepełnosprawnych” na stronie 326.
- Dokumentacja programu DB2 jest dostępna w przystępnym formacie. Więcej informacji na ten temat można znaleźć w sekcji “Dokumentacja w przystępnym formacie” na stronie 326.

Wprowadzanie danych i nawigacja za pomocą klawiatury

Operowanie programem za pomocą klawiatury

Narzędzia programu DB2 można obsługiwać za pomocą samej klawiatury. Wszystkie operacje, które można wykonać za pomocą myszy, można również wykonać za pomocą pojedynczych klawiszy lub ich kombinacji. Standardowe kombinacje klawiszy używane w systemie operacyjnym są wykorzystywane do wykonania standardowych operacji w systemie operacyjnym.

Więcej informacji o korzystaniu z klawiszy lub kombinacji klawiszy do wykonania określonych operacji można znaleźć w sekcji Skróty i akceleratory klawiszowe: Wspólny interfejs GUI - Pomoc.

Nawigacja przy użyciu klawiatury

Interfejs użytkownika narzędzi DB2 umożliwia nawigację przy użyciu klawiszy lub kombinacji klawiszy.

Więcej informacji o korzystaniu z klawiszy lub kombinacji klawiszy do nawigowania po narzędziach DB2 można znaleźć w sekcji Skróty i akceleratory klawiszowe: Wspólny interfejs GUI - Pomoc.

Miejsce aktywne dla klawiatury

W systemach operacyjnych UNIX[®] obszar aktywnego okna, w którym obsługiwane są sekwencje klawiszy, jest podświetlony.

Przystępny ekran

W narzędziach DB2 dostępne są funkcje zwiększające dostępność programu dla użytkowników o obniżonej zdolności widzenia. Takim usprawnieniem jest między innymi możliwość dostosowywania właściwości czcionek do indywidualnych potrzeb.

Ustawienia czcionek

Za pomocą notatnika Ustawienia narzędzi można wybrać kolor, rozmiar i rodzaj czcionki tekstu wyświetlanego w menu i oknach dialogowych.

Więcej informacji o określaniu ustawień czcionki można znaleźć w sekcji Zmiana czcionki menu i tekstu: Wspólny interfejs GUI - Pomoc.

Niezależność od kolorów

Zdolność rozróżniania kolorów nie jest potrzebna, aby móc korzystać ze wszystkich funkcji tego produktu.

Zgodność z rozwiązaniami technicznymi dla niepełnosprawnych

Interfejsy narzędzi DB2 zapewniają obsługę interfejsu Java Accessibility API, który pozwala na wykorzystanie razem z produktami DB2 lektorów ekranowych i innych przydatnych technologii.

Dokumentacja w przystępnym formacie

Dokumentacja dotycząca programu DB2 jest dostępna w formacie XHTML 1.0, który jest obsługiwany przez większość przeglądarek WWW. Zastosowanie formatu XHTML umożliwia wyświetlenie dokumentacji zgodnie z preferencjami wyświetlania określonymi w używanej przeglądarce. Ponadto dzięki temu można korzystać z czytników ekranu i innych rozwiązań technicznych dla niepełnosprawnych.

Diagramy składni przedstawione są w postaci dziesiętnej z kropkami. Ten format jest dostępny tylko podczas korzystania z dokumentacji elektronicznej za pomocą lektora ekranowego.

Pojęcia pokrewne:

- “Diagramy składniowe w postaci dziesiętnej z kropkami” w *Tematach wspólnych DB2*

Zadania pokrewne:

- “Skróty i akceleratory klawiszowe: Wspólny interfejs GUI - Pomoc”
- “Zmiana czcionki menu i tekstu: Wspólny interfejs GUI - Pomoc”

Uwagi

Niniejsza publikacja została przygotowana z myślą o produktach i usługach oferowanych w Stanach Zjednoczonych. Produktów, usług lub opcji opisywanych w tym dokumencie firma IBM nie musi oferować we wszystkich krajach. Informacje o produktach i usługach dostępnych w danym kraju można uzyskać od lokalnego przedstawiciela firmy IBM. Jakakolwiek wzmianka na temat produktu, programu lub usługi firmy IBM nie oznacza, że może być zastosowany jedynie ten produkt, ten program lub ta usługa firmy IBM. Zamiast nich można zastosować ich odpowiednik funkcjonalny, pod warunkiem, że nie narusza to praw własności intelektualnej firmy IBM. Jednakże cała odpowiedzialność za ocenę przydatności i sprawdzenie działania produktu, programu lub usługi pochodzących od producenta innego niż IBM spoczywa na użytkowniku.

IBM może posiadać patenty lub złożone wnioski patentowe na towary i usługi, o których mowa w niniejszej publikacji. Używanie tego dokumentu nie daje żadnych praw do tych patentów. Wnioski o przyznanie licencji można zgłaszać na piśmie pod adresem:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Zapytania dotyczące zestawów znaków dwubajtowych (DBCS) należy kierować do lokalnych działów własności intelektualnej firmy IBM (IBM Intellectual Property Department) lub wysłać je na piśmie na adres:

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

Poniższy akapit nie obowiązuje w Wielkiej Brytanii, a także w innych krajach, w których jego treść pozostaje w sprzeczności z przepisami prawa miejscowego: FIRMA INTERNATIONAL BUSINESS MACHINES CORPORATION DOSTARCZA TĘ PUBLIKACJĘ W TAKIM STANIE, W JAKIM SIĘ ZNAJDUJE "AS IS" BEZ UDZIELANIA JAKIKOLWIEK GWARANCJI (W TYM TAKŻE RĘKOJMI), WYRAŻNYCH LUB DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ ORAZ PRZYDATNOŚCI DO OKREŚLONEGO CELU LUB GWARANCJI, ŻE PUBLIKACJA NIE NARUSZA PRAW OSÓB TRZECICH. Ustawodawstwa niektórych krajów nie dopuszczają zastrzeżeń dotyczących gwarancji wyraźnych lub domniemanych w odniesieniu do pewnych transakcji; w takiej sytuacji powyższe zdanie nie ma zastosowania.

Informacje zawarte w niniejszej publikacji mogą zawierać nieścisłości techniczne lub błędy drukarskie. Informacje te są okresowo aktualizowane, a zmiany te zostaną ujęte w kolejnych wydaniach tej publikacji. IBM zastrzega sobie prawo do wprowadzania ulepszeń i/lub zmian w produktach i/lub programach opisanych w tej publikacji w dowolnym czasie, bez wcześniejszego powiadomienia.

Jakiegokolwiek wzmianki na temat stron internetowych nie należących do firmy IBM zostały podane jedynie dla wygody użytkownika i nie oznaczają, że firma IBM w jakikolwiek sposób

firmuje te strony. Materiały dostępne na tych stronach nie są częścią materiałów opracowanych dla tego produktu IBM, a użytkownik korzysta z nich na własną odpowiedzialność.

IBM ma prawo do korzystania i rozpowszechniania informacji przysłanych przez użytkownika w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

Licencjodawcy tego programu, którzy chcieliby uzyskać informacje na temat programu w celu: (i) wdrożenia wymiany informacji między niezależnie utworzonymi programami i innymi programami (łącznie z tym opisywanym) oraz (ii) wykorzystywania wymienianych informacji, powinni skontaktować się z:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Informacje takie mogą być udostępnione, o ile spełnione zostaną odpowiednie warunki, w tym, w niektórych przypadkach, uiszczenie odpowiedniej opłaty.

Licencjonowany program opisany w tym dokumencie oraz wszystkie inne licencjonowane materiały dostępne dla tego programu są dostarczane przez IBM na warunkach określonych w Umowie IBM z Klientem, Międzynarodowej Umowie Licencyjnej IBM na Program lub w innych podobnych umowach zawartych między IBM i użytkownikami.

Wszelkie dane dotyczące wydajności zostały zebrane w kontrolowanym środowisku. W związku z tym rezultaty uzyskane w innych środowiskach operacyjnych mogą się znacząco różnić. Niektóre pomiary mogły być dokonywane na systemach będących w fazie rozwoju i nie ma gwarancji, że pomiary te wykonane na ogólnie dostępnych systemach dadzą takie same wyniki. Niektóre z pomiarów mogły być estymowane przez ekstrapolację. Rzeczywiste wyniki mogą być inne. Użytkownicy powinni we własnym zakresie sprawdzić odpowiednie dane dla ich środowiska.

Informacje dotyczące produktów innych firm zostały uzyskane od dostawców tych produktów z opublikowanych przez nich zapowiedzi lub innych powszechnie dostępnych źródeł. Firma IBM nie testowała tych produktów i nie może potwierdzić dokładności pomiarów wydajności, kompatybilności ani żadnych innych danych związanych z tymi produktami. Pytania dotyczące możliwości produktów innych firm należy kierować do dostawców tych produktów.

Jakiegokolwiek wzmianki na temat kierunków rozwoju firmy IBM mogą ulec zmianie lub anulowaniu bez uprzedzenia i dotyczą jedynie ogólnych celów i założeń.

Publikacja ta zawiera przykładowe dane i raporty używane w codziennej działalności biznesowej. W celu kompleksowego zilustrowania tej działalności podane przykłady zawierają nazwy osób, firm i ich produktów. Wszystkie te nazwiska/nazwy są fikcyjne i jakakolwiek ich zbieżność z prawdziwymi nazwiskami/nazwami jest całkowicie przypadkowa.

LICENCJA NA PRAWA AUTORSKIE:

Niniejsza publikacja zawiera przykładowe aplikacje w kodzie źródłowym, ilustrujące techniki programowania w różnych systemach operacyjnych. Użytkownik może kopiować, modyfikować i rozpowszechniać te programy przykładowe w dowolnej formie bez uiszczenia opłat, w celu rozbudowy, użytkowania, handlowym lub w celu rozpowszechniania aplikacji

zgodnych z aplikacyjnym interfejsem programowym dla tego systemu operacyjnego, dla którego napisane były programy przykładowe. Programy przykładowe nie zostały gruntownie przetestowane. Firma IBM nie może zatem gwarantować lub sugerować niezawodności, użyteczności i funkcjonalności tych programów. Użytkownik może kopiować, modyfikować i rozpowszechniać te programy przykładowe w dowolnej formie bez uiszczania opłat, w celu rozbudowy, użytkowania, handlowym lub w celu rozpowszechniania aplikacji zgodnych z aplikacyjnymi interfejsami programowym IBM.

Każda kopia lub dowolna część programów przykładowych, albo też dowolna praca pochodna, musi zawierać poniższą informację o prawach autorskich:

© (*nazwa_firmy_użytkownika*) (*rok*). Części niniejszego kodu pochodzą z programów przykładowych firmy IBM Corp. © Copyright IBM Corp. *_rok_lub_lata_*. Wszelkie prawa zastrzeżone.

Znaki towarowe

Następujące nazwy są znakami towarowymi International Business Machines Corporation w Stanach Zjednoczonych i/lub innych krajach:

IBM
AIX
DataJoiner
DB2
DB2 Connect
DB2 Universal Database
Distributed Relational Database Architecture
DRDA
Informix
iSeries
Lotus
Lotus Notes
MVS
OS/390
VM/ESA
VSE/ESA
WebSphere
z/OS

Poniższe nazwy są znakami towarowymi lub zastrzeżonymi znakami towarowymi innych firm:

Java i wszystkie znaki towarowe związane z językiem Java są znakami towarowymi lub zastrzeżonymi znakami towarowymi firmy Sun Microsystems, Inc. w Stanach Zjednoczonych i/lub innych krajach.

Microsoft, Windows, Windows NT oraz logo Windows są znakami towarowymi firmy Microsoft Corporation w Stanach Zjednoczonych i/lub innych krajach.

Intel, logo Intel Inside, MMX i Pentium są znakami towarowymi firmy Intel Corporation w Stanach Zjednoczonych i/lub innych krajach.

UNIX jest zastrzeżonym znakiem towarowym The Open Group w Stanach Zjednoczonych i innych krajach.

Nazwy innych firm, produktów lub usług mogą być znakami towarowymi lub znakami usług innych firm.

Indeks

A

- ACCOUNTING_STRING, opcja użytkownika
 - poprawne ustawienia 241
- aktualizacje
 - autoryzacje 95
 - dużych obiektów (LOB) 97
 - lokalne 93
 - ograniczenia 96
 - opis 93
 - spójność referencyjna 97
 - zdalne 93
- ALTER NICKNAME, instrukcja
 - ograniczenia 35
 - opis 101
 - przykład
 - zmiana lokalnych nazwy kolumn 37
 - zmiana opcji kolumn 39
- ALTER NICKNAME, instrukcja
 - przykład
 - lokalny typ danych 55
- ALTER SERVER, instrukcja 28
 - przykład dla systemu stowarzyszonego 29, 30
- ALTER TABLE, instrukcja
 - opis dla systemów stowarzyszonych 101
- ALTER USER MAPPING, instrukcja
 - przykład dla systemu stowarzyszonego 32
- ALTER WRAPPER, instrukcja 27, 28
- analiza przekazywania do źródła
 - opis 9, 135
 - wpływ na charakterystyki pseudonimu 140
 - wpływ na charakterystyki serwera 136
 - wpływ na charakterystyki zapytania 142
- aplikacje 70
 - odwołania do obiektów źródeł danych 202
 - poziomy odseparowania 207
 - pseudonimy w 201
 - scenariusz stowarzyszenia 199
 - ustawianie opcji serwera 212
 - wpisywanie do katalogu informacji o źródle danych 203
 - żądania rozproszone 210
- aplikacje biznesowe
 - obsługiwane typy danych 301

B

- BioRS
 - obsługiwane typy danych 301
- BLAST
 - obsługiwane typy danych 301
 - obsługiwane wersje 5
 - poprawne obiekty dla pseudonimów 16
- buforowanie 173, 174, 175
 - tworzenie tabel buforowanych 181
 - usuwanie tabel buforowanych 184

C

- Centrum komend
 - korzystanie dla systemów stowarzyszonych 21
- Centrum kontroli poprawności
 - indykatory poprawności 119
 - konfigurowanie stowarzyszonych indykatów poprawności 120
 - monitorowanie poprawności stowarzyszonego pseudonimu i serwera 121
 - obraz stanu poprawności 122
- Centrum sterowania
 - interfejs dla systemów stowarzyszonych 21
- CLP (procesor wiersza komend)
 - funkcje stowarzyszone 21
- CODEPAGE, opcja 128
- COLLATING_SEQUENCE, opcja serwera
 - poprawne ustawienia 225
 - przykład 19
 - wpływ na globalną optymalizację 147
 - wpływ na możliwości przekazywania do źródła 136
- COMM_RATE, opcja serwera
 - poprawne ustawienia 225
 - wpływ na globalną optymalizację 147
- COMMENT ON, instrukcja 202
 - opis dla systemów stowarzyszonych 101
- COMMIT, instrukcja
 - tranzyt 213, 214
- CONNECTSTRING, opcja serwera
 - poprawne ustawienia 225
- CPU_RATIO, opcja serwera
 - poprawne ustawienia 225
 - wpływ na globalną optymalizację 147
- CREATE ALIAS, instrukcja
 - opis dla systemów stowarzyszonych 101
- CREATE FUNCTION (potomna lub szablon), instrukcja 60, 62
- CREATE FUNCTION MAPPING, instrukcja 59, 60, 64, 66, 67, 68, 69, 70
 - określanie nazw funkcji 66
- CREATE INDEX, instrukcja 19, 73, 74, 75, 77
 - opis dla systemów stowarzyszonych 101
- CREATE NICKNAME, instrukcja 50, 105, 110
- CREATE SERVER, instrukcja 4
- CREATE TABLE, instrukcja
 - opis dla systemów stowarzyszonych 101
- CREATE TYPE MAPPING, instrukcja 49, 51, 52, 53
- CREATE VIEW, instrukcja 206
 - opis dla systemów stowarzyszonych 101

D

- DATALINK, typ danych
 - nieobsługiwany 17

- DATEFORMAT, opcja serwera
 - poprawne ustawienia 225
- DB2 dla systemów Linux, UNIX i Windows
 - domyślna nazwa opakowania 12
 - domyślne proste odwzorowania typów 269
 - domyślne zwrotne odwzorowania typów 285
 - obsługiwane wersje 5
 - poprawne obiekty dla pseudonimów 16
 - poprawne typy serwerów 263
 - poziomy odseparowania 207
 - stowarzyszona obsługa obiektów LOB 208
- DB2 for iSeries
 - domyślna nazwa opakowania 12
 - domyślne proste odwzorowania typów 269
 - domyślne zwrotne odwzorowania typów 285
 - obsługiwane wersje 5
 - poprawne obiekty dla pseudonimów 16
 - poprawne typy serwerów 263
 - poziomy odseparowania 207
 - stowarzyszona obsługa obiektów LOB 208
- DB2 for VM and VSE
 - domyślna nazwa opakowania 12
 - domyślne proste odwzorowania typów 269
 - domyślne zwrotne odwzorowania typów 285
 - obsługiwane wersje 5
 - poprawne obiekty dla pseudonimów 16
 - poprawne typy serwerów 263
 - poziomy odseparowania 207
 - stowarzyszona obsługa obiektów LOB 208
- DB2 for z/OS and OS/390
 - domyślna nazwa opakowania 12
 - domyślne proste odwzorowania typów 269
 - domyślne zwrotne odwzorowania typów 285
 - obsługiwane wersje 5
 - poprawne obiekty dla pseudonimów 16
 - poprawne typy serwerów 263
 - poziomy odseparowania 207
 - stowarzyszona obsługa obiektów LOB 208
- DB2_MAXIMAL_PUSHDOWN, opcja serwera
 - decyzje wynikające z analizy przekazywania do źródła 142
 - poprawne ustawienia 225
 - wpływ na możliwości przekazywania do źródła 136
- db2exfmt, narzędzie
 - wyświetlanie planów dostępu 142, 152
- db2expln, narzędzie
 - wyświetlanie planów dostępu 142, 152

DBNAME, opcja serwera
poprawne ustawienia 225

definicje serwera
opcje serwera 31
opis 13
usuwanie 42
zmiana definicji wszystkich źródeł danych 30
zmiana wersji źródła danych 29
zmiana, przegląd i ograniczenia 28

DELETE, instrukcja 96
decyzje wartościujące w planie dostępu 144
opis dla systemów stowarzyszonych 101
przykłady dla systemów stowarzyszonych 115

diagnostyki
stowarzyszone 313

dialekt SQL
opis 9
wpływ na możliwości przekazywania do źródła 136

DISABLE, opcja odwzorowania funkcji
poprawne ustawienia 261

Documentum
obsługiwane typy danych 301
obsługiwane wersje 5
poprawne obiekty dla pseudonimów 16

DROP, instrukcja 71
definicje serwera 42
odwzorowania użytkowników 43
opakowania 41
opis dla systemów stowarzyszonych 101
pseudonimy 44

duży obiekt (LOB), typy danych
ograniczenia 210
operacje aktualizacji 97
wskaźniki 210

dynexpln, narzędzie
wyświetlanie planów dostępu 142, 152

E

Entrez
obsługiwane wersje 5
poprawne obiekty dla pseudonimów 16

Excel, pliki
obsługiwane typy danych 301
obsługiwane wersje 5
poprawne obiekty dla pseudonimów 16

Explain, narzędzia 169

Explain, narzędzie 165
pseudonimy 165

Extended Search
obsługiwane typy danych 301
obsługiwane wersje 5
poprawne obiekty dla pseudonimów 16

F

FOLD_ID, opcja serwera
poprawne ustawienia 225
przykład 30, 31

FOLD_PW, opcja serwera
poprawne ustawienia 225
przykład 31

funkcje wbudowane 18

funkcje zdefiniowane przez użytkownika (UDF) 18
obsługa transakcji 93
w aplikacjach systemów stowarzyszonych 70

G

globalna optymalizacja 146
omówienie 146
wpływ na charakterystyki pseudonimu 149
wpływ na charakterystyki serwera 147

GRANT, instrukcja
opis dla systemów stowarzyszonych 101
pseudonimy 202

GROUP BY, operator
decyzje optymalizujące plan dostępu 152
decyzje wartościujące w planie dostępu 144

H

HADR (usuwanie skutków awarii w środowiskach o wysokiej dostępności)
stowarzyszone 311

HMMER, źródło danych
obsługiwane typy danych 301
obsługiwane wersje 5
poprawne obiekty dla pseudonimów 16

I

IFILE, opcja serwera
poprawne ustawienia 225

IGNORE_UDT, opcja serwera
poprawne ustawienia 225

indykatory poprawności
stowarzyszone 120

Informix
domyślna nazwa opakowania 12
domyślne proste odwzorowania typów 269
domyślne zwrotne odwzorowania typów 285
obsługiwane wersje 5
poprawne obiekty dla pseudonimów 16
poprawne typy serwerów 263
poziomy odseparowania 207
stowarzyszona obsługa obiektów LOB 208

INFORMIX_LOCK_MODE, opcja serwera
poprawne ustawienia 225

INITIAL_INSTS, opcja odwzorowania funkcji
poprawne ustawienia 261

INITIAL_IOS, opcja odwzorowania funkcji
poprawne ustawienia 261

INSERT, instrukcja 95, 96
decyzje wartościujące w planie dostępu 144
opis dla systemów stowarzyszonych 101
przykłady dla systemów stowarzyszonych 113

instrukcje SQL
obsługa pseudonimów 100, 101

INSTS_PER_ARGBYTE, opcja odwzorowania funkcji
poprawne ustawienia 261

INSTS_PER_INVOC, opcja odwzorowania funkcji
poprawne ustawienia 261

IO_RATIO, opcja serwera
poprawne ustawienia 225
wpływ na globalną optymalizację 147

IOS_PER_ARGBYTE, opcja odwzorowania funkcji
poprawne ustawienia 261

IOS_PER_INVOC, opcja odwzorowania funkcji
poprawne ustawienia 261

IUD_APP_SVPT_ENFORCE, opcja serwera
poprawne ustawienia 225
przykłady 98

K

catalog
Patrz globalny katalog 219

catalog globalny 48
aktualizowanie statystyk 146, 203
opis 8
widoki zawierające informacje stowarzyszone 219

catalog narzędzi
tworzenie bazy danych 194

catalog zdalny, informacje 8

kolejności zestawiania
omówienie 136
opis 19
planowanie 19

kompensacja, opis 9

kompilator SQL
schemat blokowy analizy zapytania 133
w systemie stowarzyszonym 8

konfigurowanie źródeł danych
opcje pseudonimów 243

L

LOB (duży obiekt) typy danych
ograniczenia 210
operacje aktualizacji 97
wskaźniki 210

LOCK TABLE, instrukcja
opis dla systemów stowarzyszonych 101

LOGIN_TIMEOUT, opcja serwera
poprawne ustawienia 225

lokalne aktualizacje 93

lokalny katalog
Patrz globalny katalog 8

LONG, typy danych 57

Ł

łańcuchy
kolejności zestawiania 19

łączenia
decyzje optymalizujące plan dostępu 152
przykład żądania rozproszonego 210

M

- Microsoft Excel
 - Patrz Excel, pliki 5
- Microsoft SQL Server
 - domyślne nazwy opakowań 12
 - domyślne proste odwzorowania typów 269
 - domyślne zwrotne odwzorowania typów 285
 - obsługa Unicode 127, 128
 - obsługiwane wersje 5
 - poprawne obiekty dla pseudonimów 16
 - poprawne typy serwerów 263
 - poziomy odseparowania 207
 - stowarzyszona obsługa obiektów LOB 208
- MODULE, opcja opakowania
 - poprawne ustawienia 223
- modyfikowanie 28
 - LONG, typy danych 57
 - pseudonimy
 - lokalny typ danych 54
 - omówienie 34
 - opcje pseudonimów 38
- monitorowanie obrazów stanu 123
 - fragmenty zapytań stowarzyszonych 123, 124
 - pseudonimy i serwery 122
 - stowarzyszone pseudonimy i serwery 119, 121
- MQT (zmaterializowane tabele zapytań)
 - dodawanie do tabeli buforowanej 183
 - komponent tabeli buforowanej 179
 - na pseudonimach 140
 - ograniczenia pseudonimów 177
 - stowarzyszone
 - omówienie 173
 - usuwanie z tabeli buforowanej 183
 - włączanie pamięci podręcznej 182

N

- nazwy kolumn
 - zmiana 37
- niepełnosprawność 325
- niepodzielność
 - zachowywanie w instrukcjach 98
- nierelacyjne źródła danych
 - obsługiwane typy danych 301
 - określanie odwzorowań typów danych 17
- niezatwierdzone odczyty (UR)
 - poziomy odseparowania 207
- NODE, opcja serwera, poprawne ustawienia 225
- NUMERIC_STRING, opcja kolumny
 - poprawne ustawienia 253
 - przykład 39
 - wpływ na możliwości przekazywania do źródła 140

O

- obiekty lokalne
 - opis 100
- obiekty zdalne
 - opis 100

- obiekty zdalne (*kontynuacja*)
 - przykłady 100
- obiekty źródła danych
 - opis 15
 - poprawne typy obiektów 16
 - wykonywanie operacji 202
- obliczeniowe grupy partycji 162
- ODBC
 - domyślna nazwa opakowania 12
 - domyślne proste odwzorowania typów 269
 - obsługa Unicode 127, 128
 - obsługiwane wersje 5
 - poprawne obiekty dla pseudonimów 16
 - poprawne typy serwerów 263
 - poziomy odseparowania 207
 - stowarzyszona obsługa obiektów LOB 208
- odczyt powtarzalny (RR)
 - poziomy odseparowania (RR) 207
- odtwarzanie
 - system HADR dla źródeł stowarzyszonych 311
- odwzorowania funkcji
 - odwzorowania domyślne 59
 - odwzorowania na funkcje UDF 61
 - opcje
 - narzut funkcji 64
 - poprawne ustawienia 261
 - opis 18, 60
 - tworzenie 66
 - konkretny serwer źródła danych 69
 - konkretny typ i wersja źródła danych 68
 - konkretny typ źródła danych 67
 - usuwanie 71
 - wpływ na możliwości przekazywania do źródła 136
 - wyłączanie odwzorowań domyślnych 70
- odwzorowania typów danych
 - dla konkretnego obiektu źródła danych 54, 55
 - dla konkretnego serwera 53
 - dla konkretnego typu i wersji serwera 52
 - dla konkretnych źródeł danych 51
 - jak utworzyć 51
 - kiedy tworzyć 47
 - nieobsługiwane typy danych 47
 - nierelacyjne 50
 - opis 17
 - proste 269
 - opis 50
 - składnia 50
 - sytuacje, w których wymagane są nowe odwzorowania 49
 - w systemie stowarzyszonym 48
 - wpływ na możliwości przekazywania do źródła 136
 - zwrotne 285
 - opis 50
- odwzorowania użytkowników
 - opcje 14
 - opis 14
 - poprawne ustawienia 241
 - usuwanie 43
 - zmiana 32
- odwzorowanie funkcji 66

- ograniczenia
 - modyfikowanie pseudonimów 35
- ograniczenia informacyjne pseudonimy 185, 186
- OLE DB
 - domyślna nazwa opakowania 12
 - obsługiwane wersje 5
 - poprawne typy serwerów 263
 - poziomy odseparowania 207
- opakowania
 - nazwy domyślne 12
 - opis 11
 - usuwanie 41
 - zmiana 27
- opakowanie XML 133
- opcje
 - pseudonimy 243
- opcje kolumny
 - NUMERIC_STRING 205
 - określanie dla pseudonimów 39
 - opis 17
 - poprawne ustawienia 253
 - ustawianie 205
 - VARCHAR_NO_
TRAILING_BLANKS 205
 - wpływ na możliwości przekazywania do źródła 140
- opcje kolumny pseudonimu
 - opis 17
 - przykłady 39
- opcje opakowania
 - poprawne ustawienia 223
- opcje pseudonimów
 - określanie 38
- opcje serwera
 - dodawanie i zmiana 31
 - hierarchia 31
 - opis 13
 - optymalizowanie żądań rozproszonych 212
 - poprawne ustawienia 225
 - tyczasowe 13
 - ustawianie tymczasowych 31
 - wpływ na globalną optymalizację 147
 - wpływ na możliwości przekazywania do źródła 136
- operacje zapisu
 - Patrz aktualizacje 93
- operatory mnogościowe
 - decyzje wartościujące w planie dostępu 144
 - przykład żądania rozproszonego 210
- optymalizacja
 - wpływ na charakterystyki serwera 147
 - żądania rozproszone 212
- optymalizacja zapytania
 - opis 9
- optymalizator
 - model o kosztach stałych 146
 - opis 9
- Oracle
 - domyślne nazwy opakowań 12
 - domyślne proste odwzorowania typów 269
 - domyślne zwrotne odwzorowania typów 285
 - poprawne obiekty dla pseudonimów 16

Oracle (*kontynuacja*)
poziomy odseparowania 207
stowarzyszona obsługa obiektów
LOB 208
ORDER BY, operator
decyzje wartościujące w planie
dostępu 144

P

PACKET_SIZE, opcja serwera
poprawne ustawienia 225
paralelizm 158, 161, 162, 163, 164
stowarzyszone 157, 159
paralelizm mieszany
stowarzyszone źródła danych
omówienie 157
plan dostępu 169
przetwarzanie danych 164
włączanie 164
paralelizm między partycjami 157, 162
stowarzyszone 159, 161, 162, 163, 167
paralelizm wewnątrz partycji 157
stowarzyszone 158
stowarzyszone plany dostępu 165
parametry monitorowania
stowarzyszone 154
parametry monitorowania systemu
stowarzyszone 154
PASSWORD, opcja serwera
poprawne ustawienia 225
PERCENT_ARGBYTES, opcja
odzworowania funkcji
poprawne ustawienia 261
plan dostępu 165, 169
PLAN_HINTS, opcja serwera
poprawne ustawienia 225
przykład 31
wpływ na globalną optymalizację 147
plany dostępu 167
decyzje optymalizacyjne 152
decyzje wartościujące 144
opis 9
wydajność 152
wyświetlanie 142, 152
pliki o strukturze tabeli
obsługa Unicode 128, 129
obsługiwane typy danych 301
obsługiwane wersje 5
poprawne obiekty dla pseudonimów 16
pliki tekstowe
patrz także pliki o strukturze tabeli 5
podzapytania
przykład zapytania rozproszonego 210
poziomy odseparowania 207
predykaty
decyzje wartościujące w planie
dostępu 144
procedury zapisane w bazie
pseudonimy 205
statystyka pseudonimu 309
procesor wiersza komend (CLP)
funkcje stowarzyszone 21
proste odwzorowania typów
odzworowania domyślne 269
opis 50
Unicode 297, 298, 299, 300

protokołowanie archiwalne, patrz -
protokołowanie z przechowywaniem
protokołów 181
protokołowanie z przechowywaniem
protokołów
opis, tabele buforowane 181
przezroczysty kod DDL
długość kolumn LOB 83
obsługa transakcji 93
opis 81
tworzenie tabel zdalnych 83
usuwanie tabel zdalnych 88
zmiana tabel zdalnych 86
przykłady 118
przypisania
stowarzyszone 116
pseudonimy
dostęp do źródeł danych 201
odwołania w instrukcjach SQL 202
ograniczenia 97
opis 15
poprawne obiekty źródła danych 16
procedury zapisane w bazie 205
tworzenie
na pseudonimach 110
obiekty źródła danych 105
ustawianie opcji kolumny 205
usuwanie 44
w instrukcjach SQL 101
zmiana
lokalne nazwy kolumn 37
lokalny typ danych 54
lokalny typ danych, przykład 55
ograniczenia 35
omówienie 34
opcje kolumny 39
opcje pseudonimów 38
punkty zapisu
interfejsy API źródeł danych 98
PUSHDOWN, opcja serwera
poprawne ustawienia 225

R

RAWTOHEX, funkcja 217
reguły
semantyka przypisania
stowarzyszonego 116
REMOTE_AUTHID, opcja użytkownika
poprawne ustawienia 241
przykład 32
REMOTE_DOMAIN, opcja użytkownika
poprawne ustawienia 241
REMOTE_NAME, opcja odwzorowania
funkcji
poprawne ustawienia 261
REMOTE_PASSWORD, opcja użytkownika
poprawne ustawienia 241
przykład 32
REVOKE, instrukcja
opis dla systemów stowarzyszonych 101
rozwiązywanie problemów 313

S

scenariusze 199

SELECT, instrukcja
opis dla systemów stowarzyszonych 101
przykłady dla systemów
stowarzyszonych 110
semantyki przypisania stowarzyszonych
przykłady 118
serwer stowarzyszony 4
moduł opakowujący 11
opakowania 11
opis 4
SET PASSTHRU, instrukcja
uwarunkowania 214
SET SERVER OPTION, instrukcja
optymalizowanie zapytań
rozproszonych 212
przykład 31
ustawianie opcji tymczasowych 13
skrót klawiszowy
obsługa 325
sortowanie 19
specyfikacje indeksu
dla obiektów źródła danych 74
dla synonimów Informix 78
dla widoków 77
kiedy tabele uzyskują nowe indeksy 75
opis 19
stowarzyszone 73
wpływ na globalną optymalizację 149
spójność referencyjna 97
SQL Explain
wyświetlanie planów dostępu 142, 152
stabilność kursora (CS)
poziom odseparowania 207
stabilność odczytu (RS)
poziom odseparowania 207
statystyka
pseudonim 191, 194
statystyka pseudonimu
aktualizowanie 192
aktualizowanie, wymagania wstępne 194
wyświetlanie statusu aktualizacji 195
statystyki katalogu
wpływ na globalną optymalizację 149
statystyki stowarzyszone
aktualizowanie 191
stowarzyszone bazy danych
katalog systemowy 8
opis 7
strojenie
kolejności zestawiania 136
opcje kolumny pseudonimu 140
opcje serwera 136
Patrz także - wydajność 133
przetwarzanie zapytania 133
specyfikacje indeksu 147
statystyki katalogu 147
zmaterializowane tabele zapytań 140
strony kodowe 125, 128, 129
opis 19
Sybase
domyślne nazwy opakowań 12
domyślne proste odwzorowania
typów 269
domyślne zwrotne odwzorowania
typów 285
obsługiwane wersje 5
poprawne obiekty dla pseudonimów 16

Sybase (*kontynuacja*)
 poprawne typy serwerów 263
 poziomy odseparowania 207
 stowarzyszona obsługa obiektów
 LOB 208
 synonimy
 Informix, tworzenie specyfikacji
 indeksu 78
 SYSCAT, widoki katalogu 61, 219
 widok katalogu SYSCAT.TABLES 203
 SYSPROC.NNSTAT, procedura zapisana w
 bazie 192
 SYSSTAT, widoki katalogu 219
 system zarządzania rozproszoną bazą
 danych 3
 systemy stowarzyszone
 omówienie 3
 szablony funkcji
 opis 62
 tworzenie 62

T

tabele buforowane 177
 dodawanie zmaterializowanych tabel
 zapytań 183
 komponenty 179
 kreator 181
 opis 179
 tworzenie 181
 usuwanie 184
 usuwanie zmaterializowanych tabel
 zapytań 183
 włączanie pamięci podręcznej 182
 zdalne bazy danych, wpisywanie do
 katalogu 181
 tabele zdalne
 modyfikowanie 86
 Patrz także - przezroczysty kod DDL 81
 tworzenie 83
 usuwanie 88
 Teradata
 domyślna nazwa opakowania 12
 domyślne proste odwzorowania
 typów 269
 domyślne zwrotne odwzorowania
 typów 285
 poprawne obiekty dla pseudonimów 16
 poprawne typy serwerów 263
 poziomy odseparowania 207
 stowarzyszona obsługa obiektów
 LOB 208
 TIMEFORMAT, opcja serwera
 poprawne ustawienia 225
 TIMEOUT, opcja serwera
 poprawne ustawienia 225
 przykład 30, 31
 timestamp, parametr monitorowania 154
 TIMESTAMPFORMAT, opcja serwera
 poprawne ustawienia 225
 transakcje
 aktualizacje 93
 omówienie 91
 tranzyt
 COMMIT, instrukcja 213, 214
 obsługa obiektów LOB 210
 obsługa transakcji 93

tranzyt (*kontynuacja*)
 ograniczenia 10
 opis 10, 107
 przetwarzanie SQL 213
 SET PASSTHRU RESET, instrukcja 214
 SET PASSTHRU, instrukcja 214
 uwagi, ograniczenia 214
 tworzenie kopii zapasowej 311
 typy danych 83
 dla nierelacyjnych źródeł danych 301
 nieobsługiwane 17
 wpływ na możliwości przekazywania do
 źródła 140
 typy danych źródłowych
 wprowadzanie instrukcji SQL 217
 typy serwerów
 poprawne typy stowarzyszone 263
 typy zdefiniowane przez użytkownika (UDT)
 nieobsługiwane typy danych 17

U

ułatwienia dostępu
 opcje dla osób niepełnosprawnych 325
 Unicode 125, 127, 128, 129, 297, 298, 299,
 300
 UPDATE, instrukcja 96
 decyzje wartościujące w planie
 dostępu 144
 opis dla systemów stowarzyszonych 101
 przykłady dla systemów
 stowarzyszonych 114
 Usługi WWW
 obsługiwane typy danych 301

V

VARCHAR_NO_TRAILING_BLANKS,
 opcja kolumny
 poprawne ustawienia 253
 przykład 39
 wpływ na możliwości przekazywania do
 źródła 140
 VARCHAR_NO_TRAILING_BLANKS,
 opcja serwera
 poprawne ustawienia 225
 wpływ na możliwości przekazywania do
 źródła 136
 Visual Explain
 wyświetlanie planów dostępu 142, 152

W

WebSphere
 scenariusz 199
 WebSphere Business Integration, opakowanie
 obsługiwane typy danych 301
 widoki stowarzyszone
 przykłady 108, 206
 tworzenie 108, 206
 WITH HOLD, semantyka kursora
 dla pseudonimu 100
 w sesjach tranzytowych 107
 wydajność 135, 142, 158, 161, 162, 164,
 165, 185, 195
 kolejności zestawiania 136

wydajność (*kontynuacja*)
 kolejność zestawiania 147
 Patrz także - strojenie 133
 różnice SQL 136
 specyfikacje indeksu 149
 statystyki katalogu 149
 stowarzyszone 133, 185, 186, 191, 194,
 309
 szybkość komunikacji 147
 szybkość procesora 147
 szybkość urządzeń we/wy 147
 wskazówki dotyczące planu
 zdalnego 147
 wyzwalacze
 na pseudonimach 100

X

XML
 obsługiwane typy danych 301
 obsługiwane wersje 5
 poprawne obiekty dla pseudonimów 16

Z

zapytania
 fragmenty 9
 korzystanie z tranzytu 213
 łączenie lokalnych i zdalnych źródeł
 danych 110
 źródła danych
 pojedyncze 110
 wiele zdalnych 110
 zatwierdzanie dwufazowe
 operacje 91
 zatwierdzanie jednofazowe, operacje
 określone 91
 zdalne aktualizacje 93
 zdalne generowanie SQL 146
 zestawy znaków
 opis 19
 zmaterializowane tabele zapytań (MQT) 174,
 175
 dodawanie do tabeli buforowanej 183
 komponent tabeli buforowanej 179
 na pseudonimach 140
 ograniczenia pseudonimów 177
 stowarzyszone
 omówienie 173
 usuwanie z tabeli buforowanej 183
 włączanie pamięci podręcznej 182
 zwrotne odwzorowania typów
 odwzorowania domyślne 285
 opis 50
 Unicode 297, 298, 299, 300

Ź

źródła danych 7, 8
 aktualizowanie danych 114
 domyślne nazwy opakowań 12
 dostęp przez widoki stowarzyszone 108
 dostęp przy użyciu tranzytu 107
 kolejność zestawiania a wydajność 147
 korzystanie z tranzytu do generowania
 zapytań 213

źródła danych *(kontynuacja)*

- łączenie lokalnego i zdalnego źródła danych 110
- opis 4
- poprawne typy serwerów 263
- szybkość komunikacji a wydajność 147
- szybkość procesora a wydajność 147
- szybkość urządzeń we/wy a wydajność 147
- usuwanie danych 115
- wskazówki dotyczące planu zdalnego a wydajność 147
- wstawianie danych 113
- zapytanie do jednego źródła danych 110
- zapytanie do wielu zdalnych źródeł danych 110

Ż

- żądania rozproszone
- kodowanie 210
- optymalizacja 212

Kontakt z firmą IBM

Aby skontaktować się z firmą IBM w Stanach Zjednoczonych lub Kanadzie, zadzwoń pod numer 1-800-IBM-SERV (1-800-426-7378).

Aby uzyskać informacje o opcjach serwisu, należy zatelefonować na jeden z następujących numerów:

- W Stanach Zjednoczonych: 1-888-426-4343
- W Kanadzie: 1-800-465-9600

Aby zlokalizować biuro firmy IBM w danym kraju lub regionie, należy skorzystać z informacji umieszczonych na stronie IBM Directory of Worldwide Contacts pod adresem www.ibm.com/planetwide.

Informacje o produkcie

Informacje o programie DB2 Information Integrator można uzyskać telefonicznie lub w sieci WWW.

Mieszkańcy Stanów Zjednoczonych mogą zatelefonować na następujące numery:

- Zamawianie produktów lub uzyskiwanie ogólnych informacji: 1-800-IBM-CALL (1-800-426-2255)
- Zamawianie publikacji: 1-800-879-2755

W sieci WWW należy przejść na stronę: www.ibm.com/software/data/integration/db2ii/support.html. Na stronie tej znajdują się najnowsze informacje na temat:

- biblioteki technicznej
- zamawiania książek
- zasobów do pobrania
- grup dyskusyjnych
- pakietów poprawek
- nowości
- odsyłaczy do zasobów w sieci WWW

Komentarze do dokumentacji

Opinie klientów pomagają firmie IBM dostarczać jak najlepiej opracowane informacje. Czekamy na wszelkie komentarze na temat tej książki lub pozostałej dokumentacji programu DB2 Information Integrator. Komentarze można przekazywać w następujący sposób:

- przy użyciu internetowego formularza komentarzy od czytelników, znajdującego się pod adresem www.ibm.com/software/data/rcf;
- pocztą elektroniczną na adres comments@us.ibm.com. Pamiętaj o podaniu nazwy produktu z numerem wersji oraz tytułu książki i jej numeru PN (jeśli ma to zastosowanie). Jeśli komentarz dotyczy konkretnego tekstu, prosimy określić jego położenie (na przykład tytuł, numer tabeli lub numer strony).



SC85-0104-01

