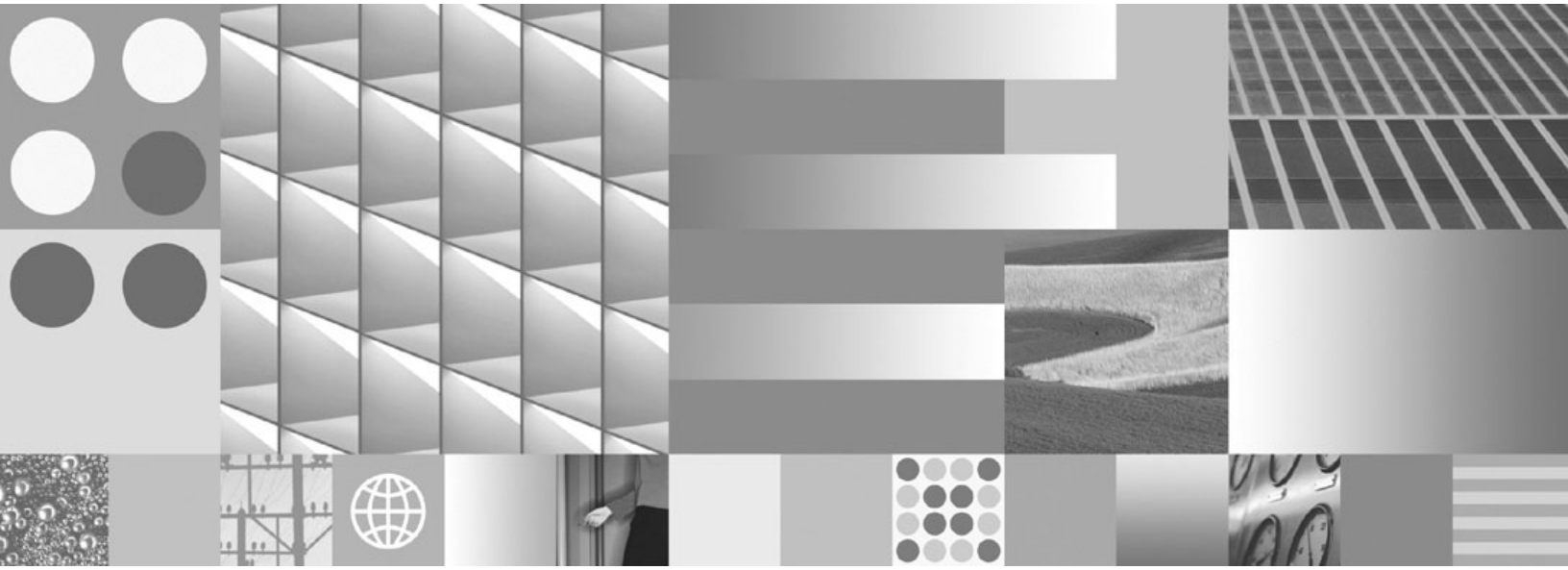
DB2 Connect wersja 9.5

Wersja 9 Wydanie 5

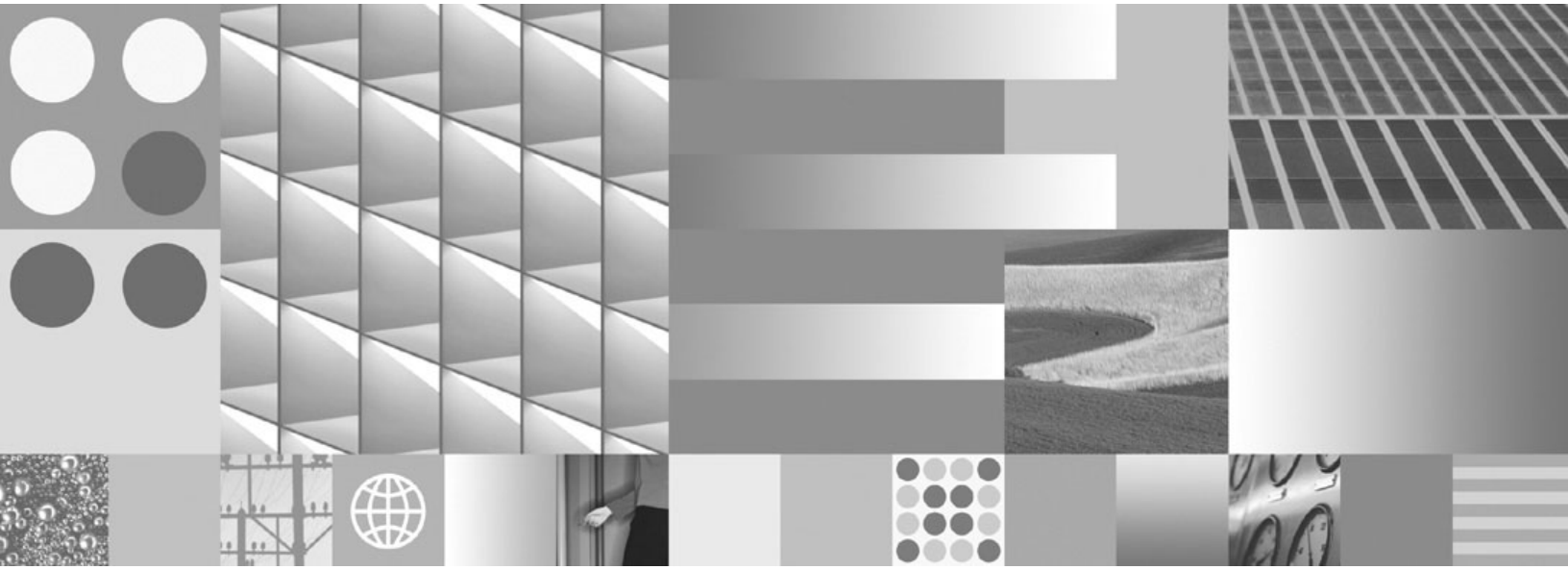


DB2 Connect - Podręcznik użytkownika
Zaktualizowano w grudniu 2010

SC85-0304-03

DB2 Connect wersja 9.5

Wersja 9 Wydanie 5



DB2 Connect - Podręcznik użytkownika
Zaktualizowano w grudniu 2010

SC85-0304-03

Uwaga

Przed skorzystaniem z tych informacji i opisywanych przez nie produktów należy przeczytać informacje ogólne, które zawiera rozdział Dodatek B, “Uwagi”, na stronie 169.

Uwagi

Niniejszy dokument zawiera informacje dotyczące produktów firmy IBM. Są one prezentowane zgodnie z warunkami umowy licencyjnej i są chronione prawem. 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 poprzez stronę WWW lub u lokalnego 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

Aby zamówić książki DB2 w firmie IBM w Stanach Zjednoczonych lub Kanadzie, należy zadzwonić do działu DB2 Marketing and Sales pod numer 1-800-IBM-4YOU (426-4968).

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 IBM Corporation 1993, 2010.

Spis treści

O tym podręczniku vii

Część 1. Pojęcia związane z programem DB2 Connect 1

Rozdział 1. DB2 Connect 3
Oferta produktów DB2 Connect 3
Funkcje wprowadzone w wersji 9 i wcześniejszych wydaniach 3
Bazy danych hosta 6
Program DB2 Connect i instrukcje SQL. 6
Narzędzia administracyjne DB2 Connect 7
Produkty WebSphere Federation Server i DB2 Connect . . 7

Rozdział 2. Architektura DRDA 9
Architektura DRDA i dostęp do danych 9
Program DB2 Connect i architektura DRDA 9
Zdalna jednostka pracy 11
Żądania rozproszone 12

Rozdział 3. Scenariusze korzystania z programu DB2 Connect 15
Bezpośredni dostęp do baz danych na hoście 15
Dostęp do danych DB2 hosta lub systemu System i za pomocą programu DB2 Connect Personal Edition . . . 17
Produkty serwerowe DB2 Connect jako serwery połączeń 18
Program DB2 Connect i aplikacje WWW 20
Program DB2 Connect i IBM WebSphere 21
DB2 Connect jako serwer aplikacji Java 21
Program DB2 Connect na serwerze WWW 22
Program DB2 Connect i serwery aplikacji 23
Program DB2 Connect i monitory przetwarzania transakcyjnego 26

Część 2. Informacje dodatkowe o produkcie DB2 Connect 29

Rozdział 4. Aktualizowanie katalogów bazy danych 31
Wartości w systemowym katalogu bazy danych 31
Wartości w katalogu węzłów. 32
Wartości w katalogu DCS 33
Arkusz dostosowania katalogu 37
Definiowanie wielu pozycji dla tej samej bazy danych . . 38
Obsługa danych dwukierunkowych. 38

Rozdział 5. Bezpieczeństwo w programie DB2 Connect 41
Zaufane połączenia za pośrednictwem produktu DB2 Connect 41
Tworzenie i kończenie zaufanego połączenia przy pomocy interfejsu CLI 42
Przełączanie użytkowników w zaufanym połączeniu poprzez interfejs CLI 43
Uwagi dotyczące uwierzytelniania w programie DB2 Connect 46
Obsługa protokołu Kerberos 47
Wskazówki dotyczące zabezpieczeń w systemach OS/390 i z/OS 48
Typy uwierzytelniania obsługiwane w programie DB2 Connect 49

Rozdział 6. Wiązanie aplikacji i programów narzędziowych (DB2 Connect) 51

Rozdział 7. Aktualizacje na wielu serwerach 55
Włączanie aktualizacji na wielu serwerach przy użyciu Centrum sterowania 56
Testowanie aktualizacji na wielu serwerach przy użyciu Centrum sterowania 56
Aktualizacja na wielu serwerach i menedżer punktów synchronizacji 57
Konfigurowanie programu DB2 Connect z menedżerem transakcji zgodnym ze standardem XA. 57
Obsługa luźno powiązanych transakcji w programie DB2 Connect 58

Rozdział 8. Przenoszenie danych przy użyciu produktu DB2 Connect 59

Rozdział 9. Odwzorowanie kodów SQLCODE 63
Wyłączanie odwzorowania kodów SQLCODE 63
Dostosowanie odwzorowania kodów SQLCODE . . . 63

Rozdział 10. Monitorowanie systemów baz danych i program DB2 Connect . . 69
Monitorowanie połączeń klientów zdalnych 69
Korzystanie z Monitora wydajności w systemie Windows 69
Korzystanie z komend GET SNAPSHOT 70
Status aplikacji DCS 72
Monitor poprawności i alerty 76
Przegląd monitora poprawności produktu DB2 for z/OS 76
Uruchamianie, zatrzymywanie i odświeżanie monitora poprawności produktu DB2 for z/OS 77
Wyświetlanie, wprowadzanie i zapisywanie zalecanych czynności 78
Wyświetlanie podsumowań alertów poprawności . . 81
Wyświetlanie obiektów alertów poprawności 82

Część 3. Wysoka dostępność i program DB2 Connect 85

© Copyright IBM Corp. 1993, 2010

Rozdział 11. Wysoka dostępność i równoważenie obciążenia w połączeniach z bazą danych na hoście . 87

Rozdział 12. Opis i konfiguracja automatycznego przekierowywania klientów (DB2 Connect) 89

Rozdział 13. Konfigurowanie automatycznego przekierowywania klientów dla technologii dystrybucji połączeń klientów 91

Część 4. Strojenie i program DB2 Connect. 93

Rozdział 14. Uwagi dotyczące wydajności programu DB2 Connect . . 95

Rozdział 15. Optymalizacja dostępu ODBC 99

Rozdział 16. Projektowanie aplikacji 101

Rozdział 17. Zarządzanie połączeniami 105
Zestawianie połączeń 105
Koncentrator połączeń 107
Pule połączeń i koncentrator połączeń 111
Menedżer transakcji WebSphere MQ i produkt DB2 for OS/390 wymagają koncentratora połączeń 112

Rozdział 18. Obsługa systemu Sysplex w serwerach DB2 Connect 113
Uwagi dotyczące wykorzystania systemu SYSPLEX w systemach OS/390 i zSeries 113
Wykorzystanie systemu Sysplex w DB2 114
Wymagania konfiguracyjne systemu Sysplex 115

Rozdział 19. Dostrajanie programu DB2 Connect. 117
Dostrajanie bazy danych hosta 118
Uwagi dotyczące dostrajania sieci 119
Rywalizacja o zasoby systemowe 120
Rozwiązywanie problemów dotyczących wydajności w programie DB2 Connect 121
Dostrajanie programu DB2 for OS/390 and z/OS . . . 121
Zwiększanie szybkości przesyłania danych przez oprogramowanie DB2 Connect 121
Dodatkowy blok zapytania 122
Skalowanie okien RFC-1323 123
Konwersja danych hosta 123
Typy danych znakowych 124
Sprzęt sieciowy 124

Rozdział 20. Dostrajanie wydajności aplikacji CLI/ODBC 127

Część 5. Rozwiązywanie problemów 129

Rozdział 21. Rozwiązywanie problemów 131
Zbieranie odpowiednich informacji 131
Połączenie początkowe nie powiodło się 131
Problemy występujące po połączeniu początkowym . . 132
Nieobsługiwane komendy DDM 133
Narzędzia diagnostyczne 134

Rozdział 22. Śledzenie DB2 w programie DB2 Connect 137
Uzyskiwanie pliku śledzenia DB2 przy użyciu narzędzia db2trc 137
Zrzut pliku śledzenia DB2 138
Formatowanie pliku śledzenia DB2 138

Rozdział 23. Pliki śledzenia DRDA. . . 141
Narzędzie śledzenia 141
Dane wyjściowe śledzenia 142
Analiza pliku danych wyjściowych śledzenia 142
Przykładowy plik danych wyjściowych śledzenia . . . 144
Informacje w kolejnych buforach dla śledzenia DRDA 148

Część 6. Komunikaty 151

Rozdział 24. Najczęściej występujące problemy z produktem DB2 Connect . 153

Część 7. Dodatki i uzupełnienia 157

Dodatek A. Przegląd informacji technicznych o programie DB2 159
Biblioteka techniczna produktu DB2 w formacie PDF lub w postaci drukowanej 160
Zamawianie drukowanych podręczników z biblioteki DB2 162
Wyświetlanie pomocy dotyczącej stanu SQL przy użyciu procesora wiersza komend 163
Uzyskiwanie dostępu do różnych wersji Centrum informacyjnego DB2 163
Wyświetlanie tematów w Centrum informacyjnym DB2 w preferowanym języku użytkownika 164
Aktualizowanie Centrum informacyjnego DB2 zainstalowanego na komputerze lokalnym lub serwerze intranetowym 164
Kursy DB2 166
Informacje dotyczące rozwiązywania problemów z programem DB2 167
Warunki 167

Dodatek B. Uwagi 169

Indeks 173

O tym podręczniku

DB2 Connect - Podręcznik użytkownika zawiera informacje na temat korzystania z produktu DB2 Connect. Przedstawiono pojęcia związane z programem DB2 Connect oraz typowe scenariusze pokazujące zależności pomiędzy programem DB2 Connect i innymi elementami środowiska sieciowego. Omówiono katalogi baz danych, bezpieczeństwo między systemami, przenoszenie danych i monitorowanie programu DB2 Connect. Przedstawiono także sposób obsługo opcji wysokiej dostępności w programie DB2 Connect. Zaprezentowano także sposoby zapewniania wysokiej wydajności środowiska sieciowego i programu DB2 Connect oraz metody rozwiązywania możliwych problemów.

Kto powinien przeczytać ten podręcznik?

Podręcznik ten przeznaczony jest dla administratorów systemu, administratorów baz danych oraz specjalistów komunikacji.

© Copyright IBM Corp. 1993, 2010

Część 1. Pojęcia związane z programem DB2 Connect

© Copyright IBM Corp. 1993, 2010

Rozdział 1. DB2 Connect

Produkt DB2 Connect zapewnia szybką i stabilną łączność z hostem i bazami danych System i dla e-biznesu oraz innymi aplikacjami działającymi pod kontrolą systemów operacyjnych Linux®, UNIX® i Windows®.

Produkt DB2 Connect Personal Edition zapewnia bezpośrednią łączność z hostem i serwerami System i DB2, podczas gdy produkty serwerowe DB2 Connect udostępniają pośrednią łączność, która pozwala klientom na dostęp do hosta i serwerów System i DB2 przez bramę DB2 Connect. Różnorodność produktów serwerowych DB2 Connect zapewnia szeroki zestaw unikalnych pakietów rozwiązań i licencji, pozwalający na wybór produktu właściwego dla konkretnego środowiska.

Oferta produktów DB2 Connect

Program DB2 Connect oferuje kilka rozwiązań z zakresu połączeń, między innymi DB2 Connect Personal Edition, a także pewną liczbę produktów serwera DB2 Connect.

- DB2 Connect Enterprise Edition
- DB2 Connect Application Server Edition
- DB2 Connect Unlimited Edition for zSeries
- DB2 Connect Unlimited Edition for iSeries

Szczegółowe informacje o opcjach produktu DB2 Connect zawiera strona www.ibm.com/software/data/db2/db2connect/

Funkcje wprowadzone w wersji 9 i wcześniejszych wydaniach

W tej sekcji przedstawiono zestawienie udoskonaleń wprowadzonych w każdej wersji i każdym wydaniu.

Funkcje wprowadzone w programie DB2 Connect wersja 9

W programie DB2 Connect wersja 9 dostępne są następujące udoskonalenia:

- Obsługa zaufanych połączeń po stronie klienta

  Klient może tworzyć zaufane połączenia (stosując technologię ODBC, interfejs XA lub nowe metody Java™) z serwerami baz danych (aktualnie tylko DB2 for z/OS), które obsługują zaufane konteksty. Nazwa użytkownika klienta może zostać wtedy zmieniona bez konieczności pełnego uwierzytelnienia nowej nazwy przez serwer bazy danych.
- Obsługa typów danych BINARY, VARBINARY i DECFLOAT

  Produkt DB2 for z/OS obsługuje teraz typy danych BINARY, VARBINARY i DECFLOAT. Obsługa tych typów danych została włączona do interfejsu DB2 CLI oraz Dostawcy serwera danych IBM dla platformy .NET. Aplikacje wykorzystujące produkt DB2 Connect do uzyskiwania dostępu do produktu DB2 for z/OS mogą używać interfejsu DB2 CLI i Dostawcy serwera danych IBM dla platformy .NET, aby skorzystać z nowych typów danych. Nowe ustawienie połączenia o nazwie SQL_ATTR_DECFLOAT_ROUNDING_MODE pozwala klientowi na określenie, jaki typ zaokrąglenia ma zostać użyty, gdy dowolne operacje po stronie serwera wymagają zaokrąglenia dziesiętnej wartości zmiennopozycyjnej.
- Protokoły komunikacyjne NetBIOS i SNA nie są już obsługiwane

© Copyright IBM Corp. 1993, 2010

Klienci korzystający z tych protokołów muszą ponownie wpisać do katalogu swoje węzły i ich bazy danych, korzystając z obsługiwanego protokołu, takiego jak TCP/IP.

- Dodana obsługa protokołu komunikacyjnego IPv6

  Dodano obsługę protokołu Internet Protocol Version 6 (IPv6), można więc nawiązywać połączenia z serwerami, korzystając zarówno z adresów protokołu IPv4, jak i IPv6.

- Usunięty limit 64 kB procesora wiersza komend (CLP) dla instrukcji SQL

  Nowy limit procesora wiersza komend (CLP) o wartości około 2 MB dla instrukcji SQL oraz dla instrukcji CLP zawierających komponenty instrukcji SQL jest porównywalny z limitami dla innych narzędzi DB2. Aplikacje używające produktu DB2 Connect mogą teraz korzystać z nowego limitu.

- Rozszerzenia Dostawcy serwera danych IBM dla platformy .NET, w tym obsługa środowiska .NET Framework 2.0

  Ta obsługa i rozszerzenia pozwolą na projektowanie doskonalszych aplikacji .NET korzystających z produktu DB2 Connect. Niektóre z nowych możliwości to:

  – Aplikacje mogą pobierać konkretny zestaw wierszy, co eliminuje konieczność przewijania pełnego zestawu wynikowego.
  – Aplikacje mogą wykonywać operację kopiowania wielkich ilości danych.
  – Aplikacje mogą określać liczbę instrukcji SQL, które wspólnie utworzą zadanie wsadowe wysyłane do serwera bazy danych DB2. Powoduje to zmniejszenie liczby pojedynczych transmisji danych pomiędzy aplikacją klienta i serwerem bazy danych.

- Zatwierdzanie dwufazowe dla źródeł danych pochodzących od wielu dostawców przy zastosowaniu serwera WebSphere Federation Server

  Aplikacje produktu DB2 Connect mogą korzystać z serwera WebSphere Federation Server w celu uzyskiwania dostępu do źródeł danych oferowanych przez wielu dostawców IBM® oraz innych niż IBM.

- Obsługa przekroczenia limitu czasu połączenia dla aplikacji bazy danych

  Można ograniczyć czas, przez który aplikacje bazy danych produktu DB2 Connect powinny oczekiwać na nawiązanie połączenia. Ta funkcja jest szczególnie przydatna w sytuacji, gdy docelowy serwer bazy danych jest niedostępny.

- Łatwiejsza aktualizacja produktu DB2 Connect Personal Edition

  Produkt DB2 Connect Personal Edition dla systemów operacyjnych Windows i Linux można zaktualizować przez dostarczenie odpowiedniego pliku elektronicznego certyfikatu. Podczas aktualizowania nie jest już konieczne wykonywanie pełnej instalacji.

- Zmiany w obsłudze licencji produktu DB2

  Zmiany w pakietach produktu DB2 Connect są częścią rozszerzeń Centrum Licencji i komendy (db2licm) narzędzia Licensed Management Tool.

Funkcje wprowadzone w programie DB2 Connect, wersja 8 wydanie 2

W programie DB2 Connect wersja 8.2 dostępne są następujące udoskonalenia:

- Automatyczne przekierowanie klienta

  Jeśli nastąpi zerwanie połączenia TCP/IP z serwerem lub programem DB2 Connect, klient automatycznie dokona próby ponownego nawiązania tego połączenia, o ile istnieje serwer alternatywny. (Serwer alternatywny jest określony na instancji serwera i jego położenie jest wysyłane do klienta podczas nawiązywania połączenia).

- Szyfrowanie danych

W komunikacji między klientem a serwerem dostępna jest teraz funkcja szyfrowania danych użytkownika przesyłanych przez sieć.

Funkcje wprowadzone w programie DB2 Connect, wersja 8 wydanie 1 (z wszystkimi pakietami poprawek i poziomami modyfikacji)

W programie DB2 Connect wersja 8.1 dostępne są następujące udoskonalenia:

- Obsługa dłuższych instrukcji SQL (o wielkości do 2 MB)

  W aplikacjach korzystających z interfejsu CLI i JDBC można używać instrukcji SQL o wielkości do 2 MB. Jednak dla osadzonego interfejsu obowiązuje ograniczenie do 64 kB.
- Informacje diagnostyczne identyfikujące pochodzenie instrukcji SQL

  Umożliwiają zidentyfikowanie aplikacji, z której pochodzi określona instrukcja w pamięci podręcznej dynamicznych instrukcji SQL programu DB2 for z/OS.
- Kolumnowa tablica wejściowa

  Pozwala aplikacjom na określenie wielu zestawów parametrów dla pojedynczej instrukcji SQL.
- Monitorowanie czasu korzystania z sieci

  Nowe elementy monitora wykorzystywane są do uzyskania lepszego obrazu operacji na bazie danych i ruchu w sieci na poziomie bazy danych lub aplikacji.
- Obsługa dynamicznego kursora przewijalnego wykorzystującego interfejs DB2 CLI

  Dynamiczne kursory przewijalne są teraz obsługiwane w ramach interfejsu DB2 CLI podczas uzyskiwania dostępu do serwerów DB2 Universal Database (UDB) for z/OS, wersja 8.1 lub nowsza.
- Obsługa eWLM

  Umożliwia monitorowanie całego zakresu jednostek roboczych w grupach oprogramowania pośredniego w celu zidentyfikowania wąskich gardeł.
- Udoskonalenia komendy ping w programie DB2

  Komenda ping produktu DB2 obsługuje teraz specyfikację wielkości pakietu żądania i odpowiedzi.

  Uwaga: W programie DB2 Connect nie jest obsługiwana komenda PING, jeśli została ona wprowadzona dla hosta w kliencie w wersji 7 za pośrednictwem gatewaya w wersji 9.

Funkcje wprowadzone w programie DB2 Connect, wersja 7 wydanie 2

W programie DB2 Connect wersja 7.2 dostępne są następujące udoskonalenia:

- poprawiona obsługa technologii Microsoft® Transaction Server (MTS) i COM+
- DB2 Connect Web Starter Kit
- Produkt DB2 Connect dla systemu Linux na platformie S/390

Funkcje wprowadzone w programie DB2 Connect, wersja 7 wydanie 1

W programie DB2 Connect wersja 7.1 dostępne są następujące udoskonalenia:

- koncentrator XA
- ulepszenia w zakresie aktualizacji na wielu serwerach

Bazy danych hosta

W niniejszym dokumencie termin *baza danych* oznacza system zarządzania relacyjnych baz danych (RDBMS, relational database management system). Inne systemy, z którymi komunikuje się program DB2 Connect, mogą używać terminu baza danych w nieco innym znaczeniu. Termin baza danych w programie DB2 Connect może dotyczyć także następujących obiektów:

OS/390 lub z/OS
DB2 Universal Database (UDB) for OS/390 and z/OS, wersja 7, lub DB2 UDB for z/OS, wersja 8. Podsystem DB2 Universal Database for z/OS and OS/390 identyfikowany na podstawie wartości LOCATION NAME. LOCATION NAME można określić, logując się do TSO i wydając następujące zapytanie SQL za pomocą jednego z dostępnych narzędzi tworzenia zapytań:

```
select current server from sysibm.sysdummy1
```

Definicja LOCATION NAME znajduje się także w zestawie danych programu startowego (BSDS), jak również w komunikacie DSNL004I (LOCATION=położenie), który jest zapisywany w momencie uruchamiania programu Distributed Data Facility (DDF). Parametr LOCATION NAME obsługuje najwyżej 8 aliasów nazw położenia, co pozwala aplikacjom na korzystanie z różnych nazw aliasów baz danych w celu uzyskania dostępu do serwera z/OS w wersji 8. Do uzyskania nazwy położenia serwera DB2, nazwy domeny, adresu IP i numeru portu służy komenda systemu z/OS -display ddf.

VSE Produkt DB2 for VSE działający na partycji bazy danych identyfikowanej przez jej nazwę DBNAME.

VM Program DB2 for VM działający na wirtualnej maszynie CMS, identyfikowany przez swoją nazwę DBNAME.

OS/400
Program DB2 for i5/OS, integralna część systemu operacyjnego OS/400. Jeśli system nie został skonfigurowany do korzystania z niezależnych pul pamięci, na serwerze System i może istnieć tylko jedna baza danych.

Program DB2 Connect i instrukcje SQL

Program DB2 Connect przekazuje instrukcje SQL wysyłane przez programy użytkowe do serwerów baz danych na hostach lub serwerach System i.

Produkt DB2 Connect może przekazać prawie wszystkie poprawne instrukcje SQL, jak również obsługiwane interfejsy DB2 API (aplikacyjne interfejsy programistyczne):

- JDBC
- SQLJ
- ADO.NET
- OLE DB
- ODBC
- Perl
- PHP
- DB2 CLI
- Wbudowany SQL

Obsługa wbudowanego SQL

Istnieją dwa rodzaje przetwarzania wbudowanego SQL: statyczny SQL i dynamiczny SQL. Statyczny SQL minimalizuje czas wykonywania instrukcji SQL, przetwarzając ją jeszcze przed wykonaniem. Dynamiczny SQL jest przetwarzany po wysłaniu instrukcji SQL do serwera bazy danych na hoście lub serwerze System i. Dynamiczny SQL jest bardziej elastyczny, lecz przeważnie wolniejszy. Decyzję, czy używać statycznego, czy dynamicznego SQL, podejmuje programista. Oba rodzaje są obsługiwane przez program DB2 Connect.

Na różnych serwerach baz danych na hoście lub serwerze System i język SQL implementowany jest na różne sposoby. Produkt DB2 Connect w pełni obsługuje wspólną wersję języka IBM SQL, jak również implementacje języka SQL produktów DB2 for OS/390 and z/OS, DB2 Server for VSE & VM (wcześniej SQL/DS) i DB2 for System i. Zaleca się używanie języka IBM SQL w celu zachowania niezależności bazy danych.

Narzędzia administracyjne DB2 Connect

Administrator DB2 Connect ma do pomocy następujące programy narzędziowe:

- Procesor CLP (Procesor wiersza komend) umożliwiający wysyłanie instrukcji SQL do bazy danych na hoście lub serwerze System i. Przesyła on instrukcje SQL do wskazanej bazy danych.
- Centrum komend DB2 udostępniające interfejs graficzny dla procesora CLP (Procesor wiersza komend).
- Narzędzia importujące i eksportujące pozwalające ładować, importować i eksportować dane z/do pliku na stacji roboczej i bazy danych na hoście lub serwerze System i. Pliki te mogą być następnie wykorzystane do importowania danych do baz danych, arkuszy i innych aplikacji działających na stacji roboczej.
- Jeśli używany jest produkt serwerowy DB2 Connect, można korzystać z przeglądarki zdarzeń i Monitora wydajności. Za pomocą przeglądarki zdarzeń można przeglądać zdarzenia wyjątków zarejestrowane przez program DB2 Connect. Za pomocą Monitora wydajności można zdalnie lub lokalnie monitorować wydajność serwerów DB2 Connect i zarządzać nią.
- Centrum sterowania DB2 pozwala na monitorowanie wszystkich aspektów serwerów DB2 Connect i na zarządzanie nimi. Pozwala również administratorom na pracę z obiektami baz danych DB2 for OS/390 lub z/OS, takimi jak tabele, widoki, pule buforów i wątki.
- Program narzędziowy monitor systemu baz danych umożliwia administratorowi systemu monitorowanie połączeń systemowych. Funkcja ta jest dostępna tylko wtedy, gdy program DB2 Connect działa jako serwer. Narzędzie to pomaga także administratorowi określać źródło błędów. Administrator systemu może kojarzyć aplikacje klienta z odpowiednimi zadaniami wykonywanymi na serwerze bazy danych na hoście lub serwerze System i.

Uwaga: We wcześniejszych wersjach graficzne narzędzia administracyjne DB2, takie jak Centrum sterowania były obsługiwane na wszystkich platformach. Począwszy od wersji 9, graficzne narzędzia administracyjne DB2 są obsługiwane tylko na platformach Windows x86, Windows x64 (AMD64/EM64T), Linux x86 i Linux AMD64/EM64T. W przypadku wszystkich platform do celów administracyjnych można używać procesora wiersza komend (CLP) DB2.

Produkty WebSphere Federation Server i DB2 Connect

Produkt WebSphere Federation Server jest oddzielnym produktem, który umożliwia dostęp do danych znajdujących się w źródłach danych pochodzących od różnych dostawców i ich integrację, podczas gdy produkt DB2 Connect umożliwia usprawnienie dostępu do dużych woluminów danych znajdujących się na istniejącym hoście i serwerach dla przedsiębiorstw.

Produkt WebSphere Federation Server pomaga integrować dane przez umożliwienie przeglądania i modyfikowania kolekcji źródeł danych, jakby była pojedynczym źródłem. Dzięki zastosowaniu tego produktu dostęp do źródła danych jest dla aplikacji wywołujących całkowicie przezroczysty. Produkt WebSphere Federation Server współpracuje z produktami serwerowymi DB2 Connect. Produkt WebSphere Federation Server umożliwia rodzimy dostęp do odczytu i zapisu w odniesieniu do produktów rodziny DB2 oraz baz danych Informix, Oracle, Sybase, Teradata i Microsoft SQL Server. Produkt WebSphere Federation Server umożliwia również dostęp do odczytu do źródeł danych nierelacyjnych oraz związanych z naukami przyrodniczymi, takich jak BLAST, Documentum, Entrez, IBM Lotus Extended Search, pliki o strukturze tabeli i XML. Produkt ten może posłużyć do formułowania zapytań dotyczących danych w systemie stowarzyszonym.

Rozdział 2. Architektura DRDA

Architektura Distributed Relational Database Architecture (DRDA) to zestaw protokołów, pozwalających na wspólną pracę wielu systemów baz danych, zarówno firmy IBM, jak i innych producentów, oraz programów aplikacji. Dowolne produkty zarządzania relacyjnych baz danych wykorzystujące architekturę DRDA można połączyć w rozproszony system zarządzania relacyjnych baz danych. Architektura DRDA koordynuje komunikację między systemami. Określa ona, jakie informacje i w jaki sposób mają być przekazywane między poszczególnymi elementami.

Jednostka pracy

Jednostka pracy (UOW, unit of work) to pojedyncza transakcja logiczna. Składa się ona z ciągu instrukcji SQL, które muszą zostać wykonane pomyślnie, aby ciąg instrukcji został uznany za wykonany. W przeciwnym razie jest on uważany za niewykonany.

Rozproszona jednostka pracy

Rozproszona jednostka pracy (DUOW, distributed unit of work), nazywana również aktualizacją na wielu serwerach, to jednostka pracy, w przetworzenie której zaangażowanych jest kilka serwerów baz danych. Jednostka DUOW ma następujące właściwości:

- W jednej jednostce pracy aktualizowany jest więcej niż jeden serwer zarządzania baz danych.
- Aplikacja kieruje rozproszeniem pracy i inicjuje zatwierdzanie.
- W jednej jednostce pracy może być wiele żądań.
- Na jedno żądanie przypada jeden serwer zarządzania baz danych.
- Zatwierdzanie transakcji jest koordynowane na wielu serwerach baz danych.

Architektura DRDA i dostęp do danych

Architektura DRDA definiuje protokoły komunikacyjne bazy danych, nie określa jednak interfejsów programistycznych, czyli interfejsów API, które powinny być używane przez programistów aplikacji. Na ogół architektura DRDA może być wykorzystywana przez aplikację do przekazywania dowolnych żądań, które mogą być wykonane na docelowym serwerze DRDA. Wszystkie spośród obecnie dostępnych serwerów DRDA mogą wykonywać żądania SQL przekazane przez aplikację za pośrednictwem produktu DB2 Connect.

Firma IBM udostępnia programistom aplikacji narzędzia do generowania żądań SQL w systemach operacyjnych Windows, UNIX i Linux. Narzędzia te stanowią część klienta DB2. Menedżer bazy danych DB2 obsługuje szereg interfejsów programistycznych: ADO.NET, JDBC, SQLJ, PHP, Perl DBI, wbudowany SQL, DB2 Call Level Interface (DB2 Call Level Interface) i OLE DB. Programiści mogą korzystać z tych funkcji API do budowania aplikacji w wielu różnych językach programowania.

Program DB2 Connect i architektura DRDA

Produkt DB2 Connect implementuje architekturę DRDA, aby obniżyć koszt i zmniejszyć złożoność dostępu do danych zapisanych na serwerach DB2 Universal Database (UDB) for iSeries, DB2 for System i, DB2 UDB for OS/390 and z/OS, DB2 for z/OS, DB2 Server for VSE & VM oraz innych serwerach baz danych zgodnych z architekturą DRDA. Dzięki

© Copyright IBM Corp. 1993, 2010

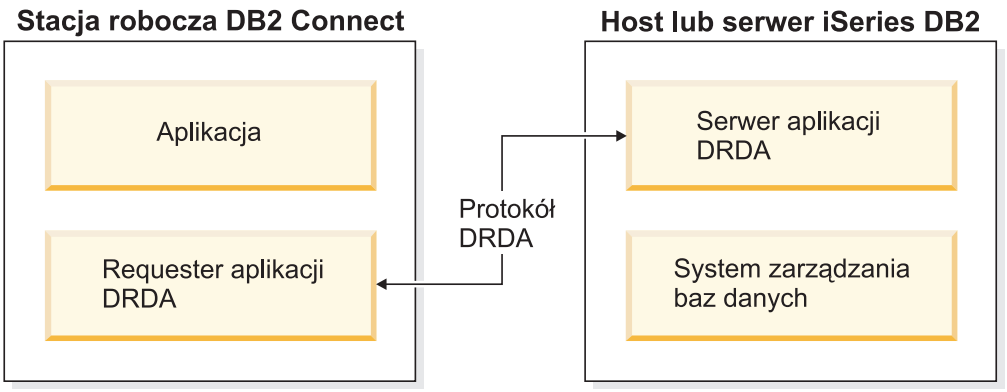
pełnemu wykorzystaniu możliwości architektury DRDA, produkt DB2 Connect oferuje wydajne, tanie rozwiązanie, charakteryzujące się wymaganymi przez klientów parametrami zarządzania systemem.

W terminologii DRDA termin *requester aplikacji (AR)* oznacza kod, który realizuje połączenie rozproszone po stronie aplikacji. Requester aplikacji to aplikacja, która żąda dostępu do danych. Produkt DB2 Connect działa jako requester aplikacji w imieniu programów, które mogą być lokalne względem stacji roboczej DB2 Connect lub położone na oddzielnym kliencie zdalnym względem produktu DB2 Connect.

Serwer aplikacji (AS, application server) to kod, który w połączeniu obsługuje bazę danych.

Architektura DRDA obsługuje również połączenia wielowarstwowe pomiędzy requesterem aplikacji i serwerem. W tej topologii serwer, z którym requester aplikacji nawiązuje połączenie, pełni rolę serwera aplikacji, lecz każdy inny serwer jest określany jako serwer bazy danych (DS, database server), ponieważ nie jest on bezpośrednio połączony z requesterem aplikacji. Ponadto, aby podkreślić, że serwer ten nie pełni roli systemu, z którego pochodzi żądanie do bazy danych, ani systemu realizującego funkcję bazy danych dla żądania, każdy serwer aplikacji lub serwer bazy danych znajdujący się między requesterem aplikacji a docelowym serwerem bazy danych jest także określany jako serwer pośredni. Korzystanie z serwerów baz danych i serwerów pośrednich obsługuje produkt DB2 Connect.

Rysunek Rys. 1 ilustruje przepływ danych między stacją roboczą DB2 Connect a hostem lub serwerem System i w przypadku, gdy istnieją tylko klienty lokalne.



Rysunek 1. Przepływ danych między serwerem DB2 Connect a hostem lub serwerem System i.

Aby zaimplementować połączenia pomiędzy systemami zarządzania bazami danych serwera DRDA i produktami IBM Data Server Client, architektura DRDA korzysta z następujących architektur:

- Character Data Representation Architecture (CDRA)
- Distributed Data Management Architecture (DDM)
- Formatted Data Object Content Architecture (FD:OCA)
- Transmission Control Protocol/Internet Protocol (TCP/IP)

Architektury te używane są jak bloki wykorzystywane w procesie budowania. Krążące w sieci strumienie danych są określone przez architekturę DRDA, która obsługuje protokół strumienia danych wspierający dostęp do rozproszonej relacyjnej bazy danych.

Żądanie zostaje skierowane do właściwego miejsca docelowego za pomocą katalogów zawierających różne typy danych komunikacyjnych oraz nazwę bazy danych serwera DRDA, do której realizowany jest dostęp.

Zdalna jednostka pracy

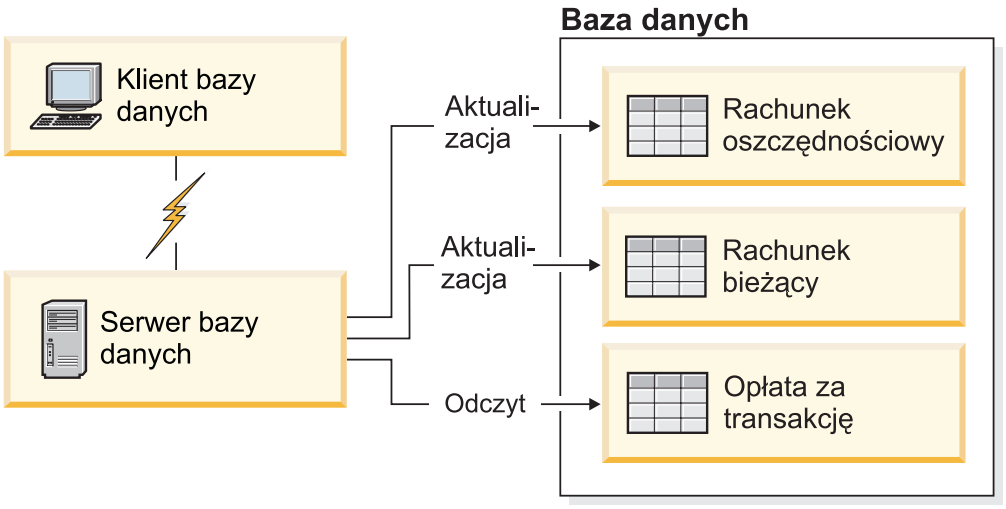
Zdalna jednostka pracy pozwala użytkownikowi lub aplikacji czytać lub aktualizować dane w jednym miejscu, w ramach pojedynczej jednostki pracy. Obsługuje ona dostęp do jednej bazy danych w obrębie pojedynczej jednostki pracy. Aplikacja może aktualizować kilka zdalnych baz danych, ale w ramach pojedynczej jednostki pracy ma dostęp tylko do jednej bazy danych.

Zdalna jednostka pracy ma następujące cechy:

- W ramach jednostki pracy obsługiwanych jest wiele żądań (instrukcji SQL).
- W ramach jednostki pracy obsługiwanych jest wiele kursorów.
- Każda jednostka pracy może aktualizować tylko jedną bazę danych.
- Aplikacja zatwierdza jednostkę pracy albo wycofuje zmiany. W wypadku niektórych błędów serwer baz danych lub program DB2 Connect może sam wycofać zmiany wprowadzone w jednostce pracy.

Rys. 2 przedstawia klienta bazy danych uruchamiającego aplikację przekazywania środków finansowych, która korzysta z bazy danych zawierającej tabele rachunku bieżącego i rachunku oszczędnościowego, jak również tabelę opłat za transakcje. Aplikacja musi:

- Zaakceptować kwotę przekazu z interfejsu użytkownika.
- Odjąć tę kwotę z rachunku oszczędnościowego i określić nowe saldo.
- Odczytać tabelę opłat bankowych, aby ustalić opłatę za transakcję dla konta o takim saldzie.
- Zmniejszyć stan rachunku oszczędnościowego o opłatę transakcyjną.
- Dodać przekazywaną kwotę do rachunku bieżącego.
- Zatwierdzić transakcję (jednostkę pracy).



Rysunek 2. Korzystanie z jednej bazy danych w transakcji.

Aby skonfigurować taką aplikację, należy wykonać następujące czynności:

1. Utwórz w tej samej bazie danych tabele dla rachunku oszczędnościowego, rachunku bieżącego oraz tabelę opłat za transakcje.

2. W wypadku zdalnej bazy danych skonfiguruj serwer bazy danych, aby korzystał z odpowiednich protokołów komunikacyjnych.
3. W wypadku zdalnej bazy danych wpisz węzeł i bazę danych do katalogu, aby zidentyfikować bazę na serwerze baz danych.
4. Dokonaj prekompilacji aplikacji, określając połączenie typu 1, czyli używając parametru CONNECT(1) w komendzie PREP.

Żądania rozproszone

Żądanie rozproszone to rozproszona funkcja bazy danych, która umożliwia użytkownikom i aplikacjom wydawanie instrukcji SQL, które w pojedynczej instrukcji odwołują się do wielu systemów zarządzania baz danych lub wielu baz danych. Na przykład złączenie tabel z dwóch różnych podsystemów DB2 for OS/390 lub z/OS.

Program DB2 Connect zapewnia obsługę żądań rozproszonych w bazach danych i systemach DBMS. Możliwe jest na przykład wykonanie operacji UNION między tabelą DB2 a widokiem Oracle. Obsługiwane systemy DBMS obejmują różne produkty z rodziny DB2 (takie jak DB2 Database for Linux, UNIX, and Windows, DB2 for OS/390 and z/OS oraz DB2 for System i), a także Oracle. Obsługa wielu dostawców jest dostępna, gdy produkt DB2 Connect jest używany razem z produktem WebSphere Federation Server.

Żądanie rozproszone zapewnia obiektom baz danych *przezroczystość położenia*. Jeśli informacje (znajdujące się w tabelach lub widokach) zostaną przeniesione, odwołania do tych informacji (zwane *pseudonimami*) mogą być aktualizowane bez wprowadzania zmian w aplikacji, która żąda tych informacji. Rozproszone żądania zapewniają także *kompensację* tym systemom zarządzania baz danych, które nie obsługują wszystkich dialektów języka SQL DB2, oraz dają pewne możliwości optymalizacji. Operacje, które nie mogą zostać wykonane w takich systemach DBMS (np. rekurencyjne operacje SQL) są uruchamiane w produkcie DB2 Connect.

Żądania rozproszone funkcjonują w sposób *półautonomiczny*. Na przykład zapytania DB2 zawierające odwołania do obiektów Oracle mogą być wprowadzane, gdy obiekty Oracle mają dostęp do tego samego serwera. Żądania rozproszone nie monopolizują ani nie ograniczają dostępu (wyjątkiem są ograniczenia wynikające z integralności i blokowania) do obiektów Oracle lub innych obiektów systemów zarządzania baz danych.

Implementacja funkcji żądania rozproszonego obejmuje instancję produktu DB2 Connect, bazę danych, która będzie służyć jako stowarzyszona baza danych, oraz jedno lub większą liczbę zdalnych źródeł danych. *Stowarzyszona baza danych* zawiera pozycje katalogu identyfikujące źródła danych oraz ich charakterystyki. *Źródło danych* składa się z systemu zarządzania baz danych oraz z danych. Aplikacje łączą się ze stowarzyszonymi bazami danych dokładnie w taki sam sposób, jak z innymi bazami danych DB2. Stowarzyszona baza danych DB2 Connect nie ma licencji do zarządzania danymi użytkowników. Jej jedyne zadanie polega na przechowywaniu informacji o źródłach danych.

Po zainstalowaniu systemu stowarzyszonego można uzyskać dostęp do informacji ze źródeł danych tak, jakby znajdowały się one w jednej dużej bazie danych. Użytkownicy i aplikacje wysyłają zapytania do stowarzyszonej bazy danych, która z kolei pobiera dane z systemów z rodziny DB2 i Oracle, zgodnie z bieżącą potrzebą. Użytkownicy i aplikacje muszą określić w zapytaniach pseudonimy, które stanowią odwołania do tabel i widoków znajdujących się w źródłach danych. Z perspektywy użytkownika pseudonimy przypominają aliasy.

Na wydajność żądań rozproszonych wpływa wiele czynników. Najważniejszy z nich polega na zapewnieniu, aby w globalnym katalogu stowarzyszonej bazy danych znajdowały się aktualne i dokładne informacje o źródłach danych oraz o znajdujących się w nich obiektach. Z

informacji tych korzysta optymalizator systemu DB2 i mogą one wpływać na przekazywanie operacji do wartościowania w źródłach danych.

Rozdział 3. Scenariusze korzystania z programu DB2 Connect

Program DB2 Connect może udostępnić szereg rozwiązań w zakresie dostępu do baz danych na hoście lub serwerze System i. W tej sekcji przedstawiono kilka scenariuszy, które mogą odpowiadać określonym potrzebom lub środowisku.

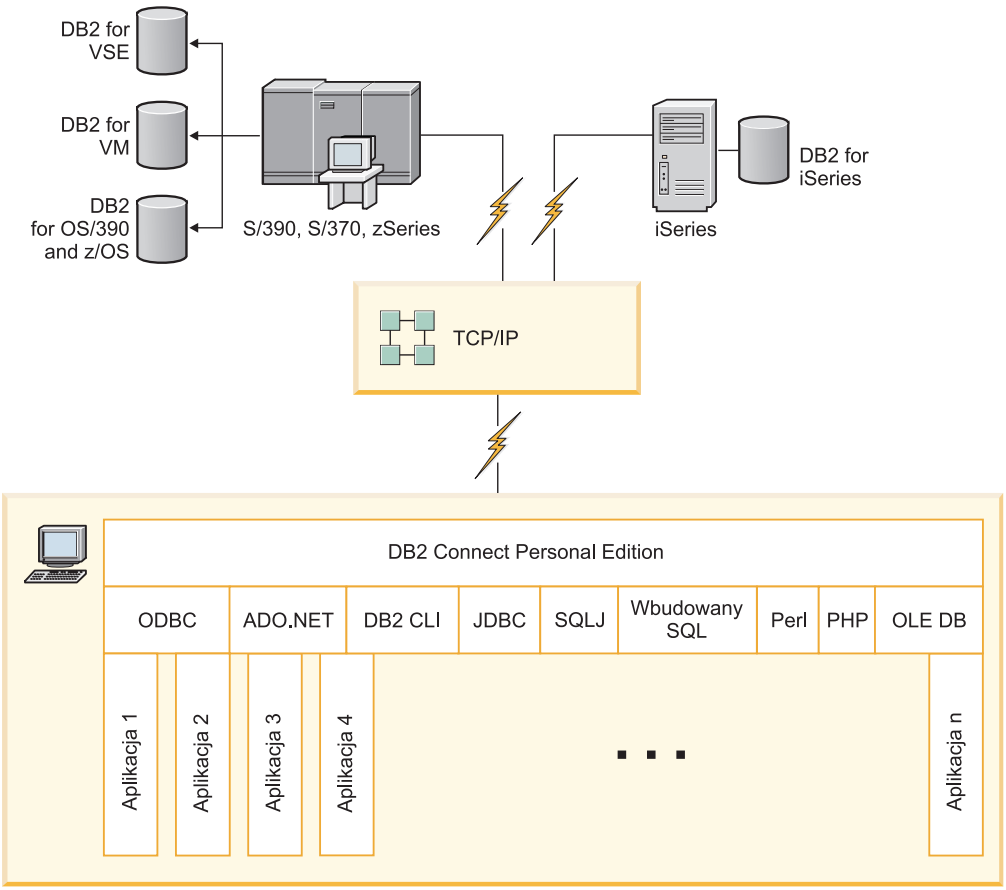
Bezpośredni dostęp do baz danych na hoście

Podstawową funkcją produktu DB2 Connect jest zapewnianie bezpośredniego połączenia z bazą danych na systemie hostowym dla akcesoriów działających na stacjach roboczych z systemem Windows, Solaris lub Linux. Wersja DB2 Connect Personal Edition oferuje najprostsze rozwiązanie tego zagadnienia.

Każda stacja robocza z zainstalowanym produktem DB2 Connect Personal Edition może nawiązać bezpośrednie połączenie TCP/IP z serwerami DB2 Universal Database (UDB) for OS/390 and z/OS, DB2 for z/OS, DB2 UDB for iSeries, DB2 for i5/OS oraz DB2 Database for Linux, UNIX, and Windows. Ponadto aplikacje mogą się łączyć z wieloma bazami danych z rodziny DB2 i aktualizować je w ramach jednej transakcji, gdzie pełną integralność danych zapewnia protokół zatwierdzania dwufazowego.

Rysunek Rys. 3 na stronie 16 ilustruje bezpośrednie połączenie stacji roboczej z zainstalowanym produktem DB2 Connect Personal Edition z bazą danych na hoście lub serwerze System i.

© Copyright IBM Corp. 1993, 2010



Rysunek 3. Połączenie bezpośrednie między programem DB2 Connect a serwerem bazy danych hosta lub systemu System i

Uwaga:

1. Program DB2 nie musi być zainstalowany na stacji roboczej DB2 Connect. Aby mieć pełny system zarządzania relacyjnych baz danych na stacji roboczej DB2 Connect, należy zamówić produkt DB2.
2. Klient IBM Data Server Client jest teraz częścią pakietu DB2 Connect i może zostać zainstalowany, jeśli użytkownik chce z niego korzystać przy projektowaniu aplikacji. Ponadto pakiet DB2 Connect zawiera teraz kreatora procedur składowanych, z którego można korzystać przy budowaniu, testowaniu i wdrażaniu procedur składowanych dla produktu DB2 for OS/390 and z/OS.
3. Programiści języka C tworzący aplikacje dla systemu Windows i korzystający z interfejsu Microsoft ODBC, OLE DB lub ADO (ActiveX Data Objects) powinni używać pakietu *Microsoft Open Database Connectivity Software Development Kit*. Programiści chcący tworzyć aplikacje z użyciem języka programowania Java mogą korzystać z dowolnego środowiska programistycznego języka Java.
4. Jeśli połączenie z serwerem bazy danych DB2 for z/OS z włączoną obsługą systemu Sysplex zostanie zerwane, klient automatycznie podejmie próbę ponownego nawiązania połączenia.

Dostęp do danych DB2 hosta lub systemu System i za pomocą programu DB2 Connect Personal Edition

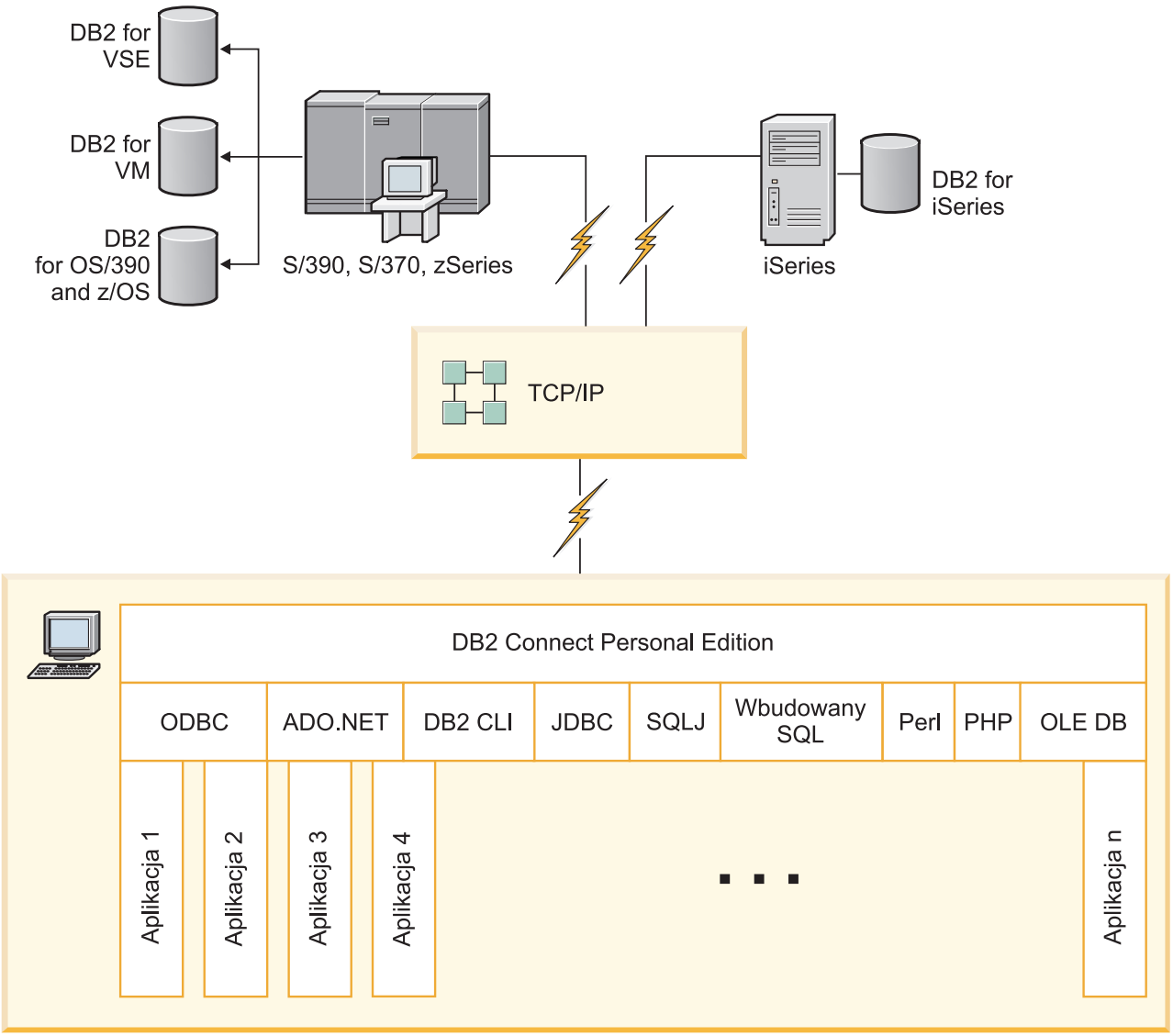
Połączenie bezpośrednie, bez serwerów pośrednich, jest niezwykle dogodną i pożądaną konfiguracją. Jest to szczególnie istotne, gdy serwer bazy danych hosta lub systemu System i obsługuje połączenia TCP/IP. W takim wypadku każda stacja robocza DB2 Connect nawiązuje bezpośrednie połączenie z hostem lub serwerem baz danych systemu System i.

Komunikacja TCP/IP wymaga, aby baza danych hosta lub systemu System i obsługiwała protokół TCP/IP. Rodzime połączenia TCP/IP są obsługiwane w następujących wersjach:

- DB2 Universal Database (UDB) for OS/390 and z/OS, wersja 7.1 lub nowsza
- DB2 UDB for iSeries, wersja 5 wydanie 1 lub nowsze
- DB2 Server for VSE & VM, wersja 7 lub nowsza.

Aby nawiązać połączenie z serwerem baz danych hosta IBM lub systemu System i, należy użyć licencjonowanego produktu DB2 Connect. Nie można ustanowić bezpośredniego połączenia z serwerem danych hosta IBM lub systemu System i przy użyciu klienta IBM Data Server Client.

Rys. 4 na stronie 18 przedstawia stację roboczą z zainstalowanym programem DB2 Connect Personal Edition bezpośrednio połączoną z serwerem bazy danych hosta lub systemu System i.

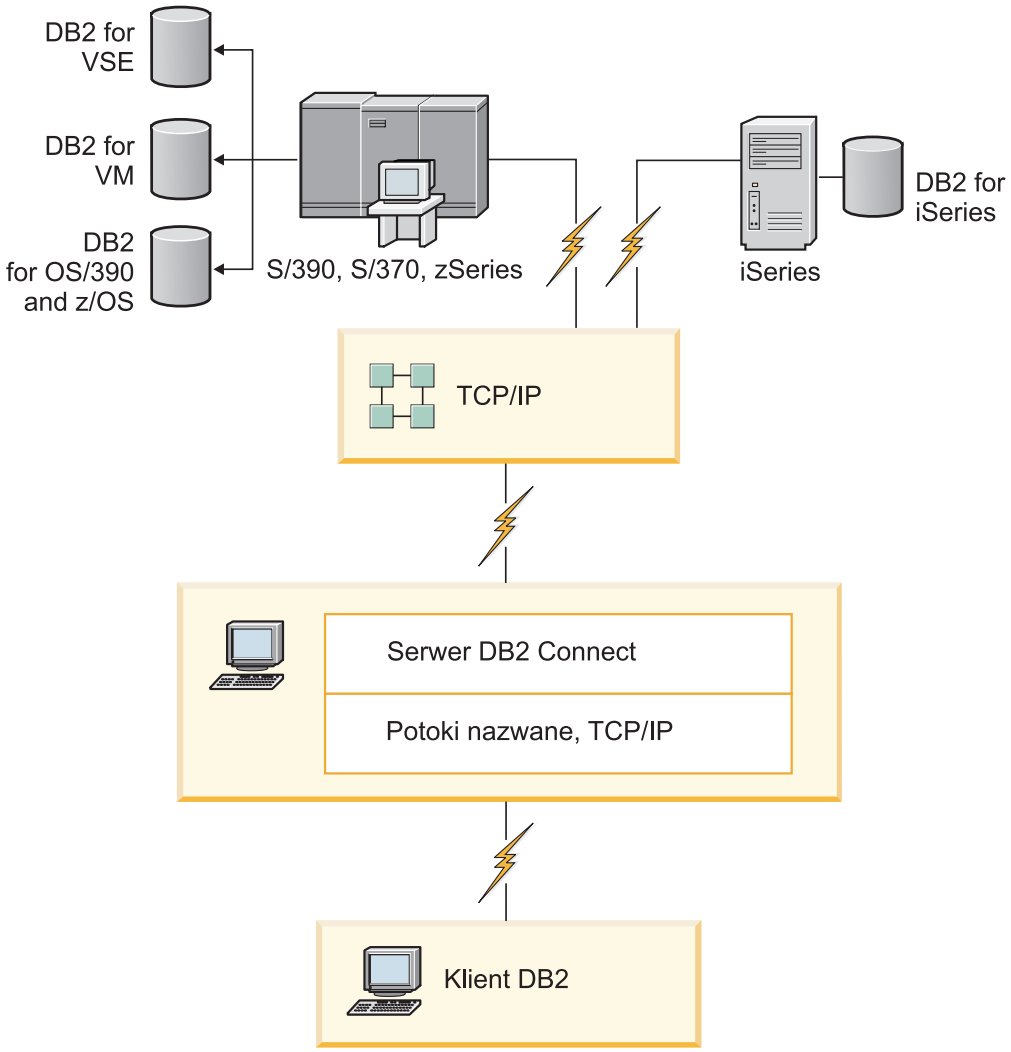


Rysunek 4. Połączenie bezpośrednie między programem DB2 Connect a serwerem bazy danych hosta lub systemu System i

Produkty serwerowe DB2 Connect jako serwery połączeń

Serwer DB2 Connect umożliwia łączenie się wielu klientów z danymi znajdującymi się na hoście lub serwerze System i. Może znacznie zredukować nakład pracy związany z zapewnieniem i utrzymaniem dostępu do danych korporacyjnych. Na rysunku Rys. 5 na stronie 19 przedstawiono opracowane przez firmę IBM rozwiązanie dla środowisk, w których klient DB2 łączy się pośrednio z hostem lub serwerem bazy danych System i za pośrednictwem produktu serwerowego DB2 Connect, takiego jak DB2 Connect Enterprise Edition.

Uwaga: Połączenia pośrednie są obsługiwane jedynie w przypadku klientów DB2 lub klientów JCC działających w systemach Linux, UNIX lub Windows. Próba nawiązania połączenia z hostem lub serwerem bazy danych System i za pośrednictwem serwera DB2 Connect przy użyciu innego klienta kończy się błędem SQL1334.



Rysunek 5. DB2 Connect Enterprise Edition

Jeśli nastąpi zerwanie połączenia TCP/IP z serwerem DB2 Connect, klient automatycznie podejmie próbę ponownego nawiązania tego połączenia. Najpierw klient spróbuje ponownie nawiązać połączenie z pierwotnym serwerem. Jeśli ponowne nawiązanie połączenia nie powiedzie się, klient dokona przełączenia awaryjnego i nawiąże połączenie z alternatywnym serwerem DB2 Connect. (Serwer alternatywny jest określony na instancji serwera i jego położenie jest zwracane do klienta podczas nawiązywania połączenia). Jeśli ponowne nawiązanie połączenia z serwerem alternatywnym nie powiedzie się, klient dokona próby ponownego nawiązania połączenia z pierwotnym serwerem. Klient będzie ponawiał próby nawiązania połączenia, przełączając się między serwerem pierwotnym a serwerem alternatywnym, aż do czasu nawiązania połączenia lub osiągnięcia określonego limitu czasu dla prób nawiązania połączenia.

Program DB2 Connect i aplikacje WWW

Przeglądarka WWW staje się coraz częściej standardowym interfejsem dla wszelkiego rodzaju aplikacji, od katalogów elektronicznych po aplikacje intranetowe. W wypadku prostych aplikacji WWW wystarczy sam serwer WWW. Dla rozbudowanych aplikacji, które wymagają dostępu do bazy danych i przetwarzania transakcyjnego, firma IBM oferuje rozwiązania wykorzystujące program DB2 Connect i obsługujące wiele transakcji jednoczesnych przez sieć WWW.

Zalety i wady tradycyjnego programowania CGI

Aplikacje e-biznesowe w sieci WWW wykorzystują zwykle interfejs Common Gateway Interface (CGI) w celu umożliwienia użytkownikom wysyłania zapytań do baz danych. W wielu firmach aplikacje sieci WWW wykorzystywane są na wewnętrzne potrzeby firmy i także mają zwykle obsługę baz danych w tle.

Użytkownicy wypełniają formularze na stronach WWW, następnie formularze te wysyłane są za pośrednictwem interfejsu CGI do aplikacji lub skryptów na serwerze WWW. Serwer ze swojej strony używa własnego interfejsu API w celu wprowadzenia zapytania SQL do bazy danych hosta. Ten sam skrypt może następnie utworzyć stronę WWW (HTML) zawierającą wyniki zapytania, po czym wysłać je z powrotem w celu wyświetlenia w przeglądarce WWW użytkownika. Przykładem może być elektroniczny katalog sklepu internetowego, w którym użytkownik sprawdza dostępność i cenę określonego towaru lub usługi.

Aplikacje CGI mogą być proste do zaprojektowania i łatwe w obsłudze. Jako że standard CGI jest niezależny zarówno od systemu operacyjnego, jak i od języka programowania, dostępny jest on na prawie wszystkich platformach komputerowych. Programy CGI mogą być napisane w języku C++ lub w języku skryptów, takim jak na przykład Perl lub PHP.

Chociaż interfejs CGI może wydawać się idealnym rozwiązaniem dla aplikacji WWW, ma on jednak istotne wady. Środowisko programowania dla CGI nie jest tak wyszukane, jak inne interfejsy API. Ponadto w przypadku prowadzenia handlu elektronicznego na dużą skalę problemem staje się skalowalność. Za każdym razem, gdy wywoływana jest aplikacja CGI, na serwerze sieci WWW tworzony jest nowy proces. Każdy proces musi utworzyć własne połączenie z bazą danych i wysłać własne zapytanie. W środowiskach o dużych liczbach transakcji ograniczenie to może być powodem powstawania poważnych problemów dotyczących wydajności.

Serwer DB2 Connect w połączeniu z serwerem WWW może być używany do tworzenia dużych i stabilnych aplikacji z zakresu handlu elektronicznego. Program DB2 Connect udostępnia kilka rozwiązań umożliwiających poprawienie wydajności aplikacji WWW. Procedury składowane umożliwiają użytkownikom DB2 Connect zmniejszenie liczby zapytań wysyłanych do bazy danych.

Grupowanie połączeń w pule zmniejsza ponadto częstotliwość połączeń i odłączeń od bazy danych.

Użycie języka PHP jako modułu lub wtyczki serwera WWW

Chociaż język PHP można wykorzystywać do programowania skryptów CGI, jest on powszechnie stosowany jako moduł lub wtyczka serwera WWW. W wieloprocesowym serwerze WWW, takim jak Apache, sterownik IBM DB2 dla PHP można wykorzystać w celu łagodzenia problemów związanych ze skalowalnością. W wieloprocesowym serwerze WWW pula procesów jest wykorzystywana ponownie do obsługi żądań serwera WWW. Aby wyeliminować konieczność tworzenia połączenia z bazą danych dla każdego żądania serwera WWW, można utworzyć połączenie stałe. W takim środowisku połączenie stałe może istnieć

poza zasięgiem pojedynczego skryptu PHP. Połączenie stałe zostanie wykorzystane ponownie, jeśli kolejne żądanie serwera WWW będzie wymagać identycznego połączenia.

Program DB2 Connect i IBM WebSphere

Produkt IBM WebSphere udostępnia bardziej kompletne rozwiązanie dla e-biznesu, niż jest to możliwe przy użyciu bardziej tradycyjnych narzędzi skryptowych, takich jak PHP. Serwery aplikacji WebSphere nie tylko realizują skryptowe możliwości PHP, lecz pozwalają także użytkownikowi na udostępnianie za pośrednictwem sieci WWW złożonych i najlepszych w swojej klasie usług, korzystanie z serwletów, stron Active Server Pages (ASP) i komponentów JavaBeans™ oraz obejmują obsługę technologii z interfejsem WWW, między innymi takich jak Java, TCP/IP, HTTP, HTTPS, HTML, DHTML, XML, MIME, SMTP, IIOP i X.509. Dzięki rozwiązaniom WebSphere można:

- wykorzystywać standardy przemysłowe do przyspieszenia prac programistycznych i poszerzenia możliwości współdziałania;
- implementować technologie narzędzi i struktury aplikacyjne innych firm;
- analizować wydajność serwisów WWW i stopień wykorzystania ich zawartości;
- w łatwy sposób skalować serwisy, aby umożliwić obsługę większej liczby użytkowników i utrzymać wysoką przepustowość;
- wdrażać w większości głównych środowisk operacyjnych (AIX, HP-UX, Linux, Novell NetWare, OS/390, z/OS, OS/400, system operacyjny Solaris, Microsoft Windows);
- korzystać z istniejących serwerów WWW, w tym serwerów firm Apache, IBM, Netscape i Microsoft.

WebSphere nie jest pojedynczym produktem, lecz rodziną trzech produktów kierowanych na trzy różne rynki docelowe. Sercem rozwiązania WebSphere jest produkt WebSphere Application Server.

Produkt WebSphere Application Server udostępnia środowisko dla trzech typów obiektów. Pierwszym z nich jest technologia Java Server Pages, analogiczna do technologii Active Server Pages. Drugi komponent składa się z serwletów Java, a trzeci to korporacyjne komponenty JavaBean. Korporacyjne komponenty JavaBean stanowią coraz bardziej znaczący standard wdrażania wielkoskalowych, stabilnych aplikacji korporacyjnych.

Aplikacje WebSphere mogą być wdrażane na tej samej platformie, co serwer WWW i baza danych DB2. W przypadku produktów DB2 Universal Database (UDB) for OS/390 and z/OS, DB2 for z/OS, DB2 for VM, DB2 for VSE, DB2 UDB for iSeries oraz DB2 for i5/OS produkt WebSphere jest wdrażany na tej samej platformie, co produkt serwerowy DB2 Connect.

Istnieje kilka rozwiązań WebSphere, podobnie jak w przypadku środowiska Rational Application Developer (RAD). Bardziej szczegółowe informacje można znaleźć pod adresem http://www.ibm.com/software/webservers/appserv/was/

DB2 Connect jako serwer aplikacji Java

Wiele z niedoskonałości języków skryptowych można wyeliminować, stosując zamiast nich język Java. Firma IBM udostępnia zarówno aplety, jak i aplikacje, w których można korzystać z języka Java na dowolnym etapie transakcji WWW. Rozwiązania oferowane przez firmę IBM umożliwiają łączenie technik, co oznacza, że z bazą danych DB2 można stosować rozwiązania skryptowe, takie jak Perl DBI lub Microsoft Active Server Pages, lub też można zwrócić się w stronę bardziej stabilnej implementacji oferowanej przez serwer aplikacji Java, taki jak produkt IBM WebSphere.

Dla programistów języka Java przeznaczone są dwa interfejsy API. Pierwszy z nich, JDBC, jest obsługiwany, aby umożliwić użycie języka Java do rozwoju korzystających z danych apletów Java i aplikacji Java, jak również serwletów Java, stron Java server pages (JSP) i komponentów Enterprise Java Bean (EJB). JDBC jest interfejsem API poziomu wywołania lub wywoływania metod. Drugi interfejs API Java to SQLJ. Interfejs SQLJ udostępnia możliwość wstawiania instrukcji SQL wewnątrz programu napisanego w języku Java. Produkt DB2 może używać obu interfejsów API w transakcji WWW, zarówno po stronie klienta, jak i serwera.

Po stronie klienta obsługiwane są aplety, aplety korzystające z danych oraz aplikacje. Po stronie bazy danych obsługa Java składa się z obiektów bazy danych, takich jak funkcje zdefiniowane przez użytkownika i procedury składowane.

W przypadku baz danych DB2 for OS/390 and z/OS, DB2 for VSE and VM, DB2 Universal Database (UDB) for iSeries oraz DB2 for i5/OS istnieją dwa różne sposoby wdrażania aplikacji Java. Można zastosować bezpośrednie połączenia udostępniane z protokołem TCP/IP przez produkt DB2 Connect Personal Edition, bądź wybrać korzystanie z produktu serwerowego DB2 Connect, który zapewnia łączność z hostem lub serwerem danych System i.

W obu tych przypadkach użytkownik pracujący w sieci WWW nie musi mieć żadnego specjalnego oprogramowania, aby uzyskać dostęp do bazy danych; wystarczy do tego standardowa przeglądarka WWW. Jedynymi elementami, które muszą zostać zainstalowane, są produkt serwerowy DB2 Connect i dowolny serwer WWW w standardzie przemysłowym. Jeśli serwer WWW i serwer DB2 Connect znajdują się fizycznie na innych komputerach, to klient IBM Data Server Client musi być zainstalowany na serwerze WWW.

W przypadku baz danych DB2 for OS/390 and z/OS kluczowym komponentem jest produkt serwerowy DB2 Connect uruchomiony na serwerze pośredniczącym. Ten komponent udostępnia na serwerze obsługę interfejsu JDBC, w uzupełnieniu do możliwości łączenia się z serwerem DB2 for OS/390 and z/OS, DB2 for VSE and VM, DB2 Universal Database (UDB) for iSeries oraz DB2 for i5/OS. Podobnie jak w powyższym przypadku, klient nie musi mieć żadnego specjalnego oprogramowania poza przeglądarką WWW.

Firma IBM zapewnia wsparcie i narzędzia dla rozwoju aplikacji i apletów Java. Na potrzeby projektowania aplikacji bazodanowych produkt DB2 Database Enterprise Developer Edition udostępnia narzędzia Rational Web Developer, DB2 Embedded Application Server, Cloudscape w wersji 10.2, a także DB2 i DB2 Connect do celów testowych. Narzędzia pochodzące od innych firm, takie jak NetBeans, Borland JBuilder lub Symantec Visual Cafe, będą również działać z rozwiązaniami bazodanowymi firmy IBM.

Program DB2 Connect na serwerze WWW

Firma IBM oferuje serwery HTTP (WWW) z wszystkimi produktami DB2 Connect. Produkty serwerowe DB2 Connect, takie jak DB2 Connect Enterprise Edition, udostępniają natychmiastową obsługę serwerów WWW Apache i Lotus Domino Go, a także mogą pracować z dowolnym innym serwerem WWW, takim jak Microsoft Internet Information Server lub Netscape Enterprise Server.

Przy pracy z bazą danych z rodziny DB2 działającą w systemie zSeries, System i, VM lub VSE, produkt serwerowy DB2 Connect jest wymagany na serwerze WWW. Produkty serwerowe DB2 Connect będą udostępniać biblioteki i interfejsy komunikacyjne, aby umożliwić serwerom WWW dostęp do tego hosta i platform System i. Do komunikacji pomiędzy serwerem WWW a bazą danych działającą w systemie zSeries, System i, VM lub VSE można wykorzystać protokół TCP/IP.

Uwaga: Rozwiązania WWW firmy IBM dają możliwość współpracy z wieloma bazami danych w tym samym skrypcie CGI (na przykład w języku PHP) lub w ramach tej same transakcji w skrypcie CGI.

Procedury składowane

Ważnym czynnikiem, który trzeba uwzględniać w aplikacjach WWW oraz w środowiskach klient/serwer, jest potrzeba minimalizacji ruchu odbywającego się między serwerem HTTP i końcową bazą danych. Czynnik ten jest szczególnie ważny, jeśli przetwarzane są duże liczby transakcji, co jest charakterystyczne dla większości aplikacji handlu elektronicznego.

Zalecane podejście polega na połączeniu programowania aplikacji za pomocą interfejsu CGI z logiką programowania i logiką biznesową, zawartą w procedurach składowanych. Produkty DB2 Database for Linux, UNIX, and Windows i DB2 Universal Database (UDB) for OS/390 and z/OS, DB2 for z/OS, DB2 UDB for iSeries, DB2 for i5/OS oraz DB2 for VSE stosują tę samą konwencję w zakresie parametrów wywołania procedur składowanych.

Podobnie jak w przypadku zwykłych skryptów interfejsu WWW przeglądarka WWW wysyła formularz do serwera WWW, na którym uruchomiony jest skrypt interfejsu WWW. Jednak zamiast wysyłania do bazy DB2 poszczególnych instrukcji SQL, wysyłane jest żądanie wykonania procedury składowanej. Taka procedura składowana zawiera pewną liczbę instrukcji języka SQL, które w innej sytuacji zostałyby uruchomione oddzielnie. Procedury składowane zmniejszają liczbę komunikatów przepływających w obie strony między skryptem interfejsu WWW i bazą danych zaplecza.

Główną korzyścią płynącą z wykorzystania procedur składowanych jest zmniejszenie ruchu w sieci pomiędzy serwerem HTTP i zapleczem bazy danych DB2.

Program DB2 Connect i serwery aplikacji

Rozwój modelu klient/serwer umożliwił projektantom aplikacji usprawnienie ich obsługi oraz obniżenie kosztów szkoleń, dzięki udostępnieniu aplikacji z graficznym interfejsem użytkownika dla platform, takich jak Windows. Jednocześnie pozwoliło to na elastyczne oddelegowanie funkcji zarządzania bazami danych do niezawodnych serwerów baz danych działających pod kontrolą różnych systemów operacyjnych i na różnych platformach sprzętowych.

Model klient/serwer, w którym logika aplikacji rozproszona jest na stacjach roboczych klientów, zwany jest popularnie *dwuwarstwową technologią klient/serwer*. W modelu dwuwarstwowym aplikacja jest umieszczona w warstwie klienta, natomiast serwer bazy danych implementuje serwer lub warstwę postprocesora. Produkt DB2 Connect zapewnia pełną obsługę dwuwarstwowych aplikacji typu klient-serwer, dla których serwerami baz danych są produkty DB2 Universal Database (UDB) for OS/390 and z/OS, DB2 for z/OS, DB2 UDB for iSeries, DB2 for i5/OS oraz DB2 for VM and VSE.

Wraz ze wzrostem wielkości aplikacji klient/serwer okazało się, że dwuwarstwowy model klient/serwer ma poważne ograniczenia. Rozpowszechnianie dużej ilości logiki biznesowej na setkach lub tysiącach klienckich stacji roboczych spowodowało, że zarządzanie zmianami stało się zadaniem skomplikowanym i kosztownym. Każda zmiana zasad biznesowych wymaga wymiany części aplikacji związanej z klientem. Bardzo często taka wymiana aplikacji musi być wykonana w tym samym czasie na wszystkich klienckich stacjach roboczych w przedsiębiorstwie, aby mieć pewność, że zasady biznesowe będą stosowane w spójny sposób.

Inne ograniczenie dwuwarstwowego modelu klient/serwer staje się oczywiste wraz ze wzrostem skali zasobów pochłanianych przez takie aplikacje. Rozmieszczanie setek lub

tysięcy *klientów pełnowymiarowych*, jak nazywa się czasami klienty z modelu dwuwarstwowego, zwiększa wymagania dotyczące mocy przetwarzania i pojemności każdej klienckiej stacji roboczej. Ponadto zwiększają się także wymagania dotyczące serwera baz danych w związku z tym, że każdy klient wymaga dedykowanego połączenia z bazą danych oraz zasobów związanych z koniecznością obsługi takiego połączenia. O ile zależność dwuwarstwowego modelu klient/serwer od rozpowszechniania logiki biznesowej można w pewnym stopniu zredukować, korzystając z procedur składowanych, to jednak pozostałych ograniczeń nie można w łatwy sposób rozwiązać bez wprowadzania zmian w modelu.

Rozwiązanie dla serwera aplikacji

W miarę wzrostu kosztów oraz stopnia złożoności dwuwarstwowych aplikacji klient/serwer większość dużych aplikacji zaczęto przekształcać z uwzględnieniem wielowarstwowego modelu klient/serwer. W modelu wielowarstwowym rola warstwy bazy danych pozostaje niezmieniona. Jednak warstwa klienta jest uzupełniana przez jedną lub kilka warstw pośrednich; zwykle jest to jedna warstwa pośrednia, stąd też model ten nosi nazwę *trójwarstwowego*.

W modelu trójwarstwowym klient obsługuje komunikację z użytkownikiem i nie zawiera żadnej logiki biznesowej. Warstwa pośrednia składa się z jednego lub kilku serwerów aplikacji. Zadaniem serwera aplikacji jest dostarczanie stabilnej i wydajnej pod względem kosztów implementacji logiki, leżącej u podstaw procesów i zasad biznesowych. Podobnie jak w przypadku modelu dwuwarstwowego, implementacja zasad biznesowych jest często uzupełniana o wykorzystywanie procedur składowanych po to, aby poprawić wydajność.

Ponieważ na stacjach roboczych nie implementuje się już całej logiki aplikacji, a stacje robocze zajęte są tylko obsługą użytkowników, wymagania dotyczące zasobów są dla warstwy klienta znacznie zmniejszone. Dlatego też warstwa klienta w modelu trójwarstwowym nazywana jest często *cienkim klientem*. Ponadto, ponieważ scentralizowany serwer aplikacji obsługuje żądania pochodzące od wszystkich klientów, ma on zdolność współużytkowania zasobów, takich jak połączenia z bazą danych między wszystkimi klientami. W związku z tym serwer baz danych nie musi już dłużej obsługiwać dedykowanych połączeń z każdym użytkownikiem aplikacji.

Obecnie dostępnych jest wiele przykładów trójwarstwowych serwerów aplikacji. Prawie wszyscy dostawcy systemów zarządzania zasobami przedsiębiorstwa (ERP, Enterprise Resource Planning) implementują swoje aplikacje w postaci modeli trójwarstwowych, takich jak na przykład aplikacje SAP R/3 lub PeopleSoft V7. Inne przykłady dotyczą wiodących dostawców systemów zarządzania związkami w przedsiębiorstwie, takich jak Siebel i Vantive.

Serwery aplikacji i DB2 Connect

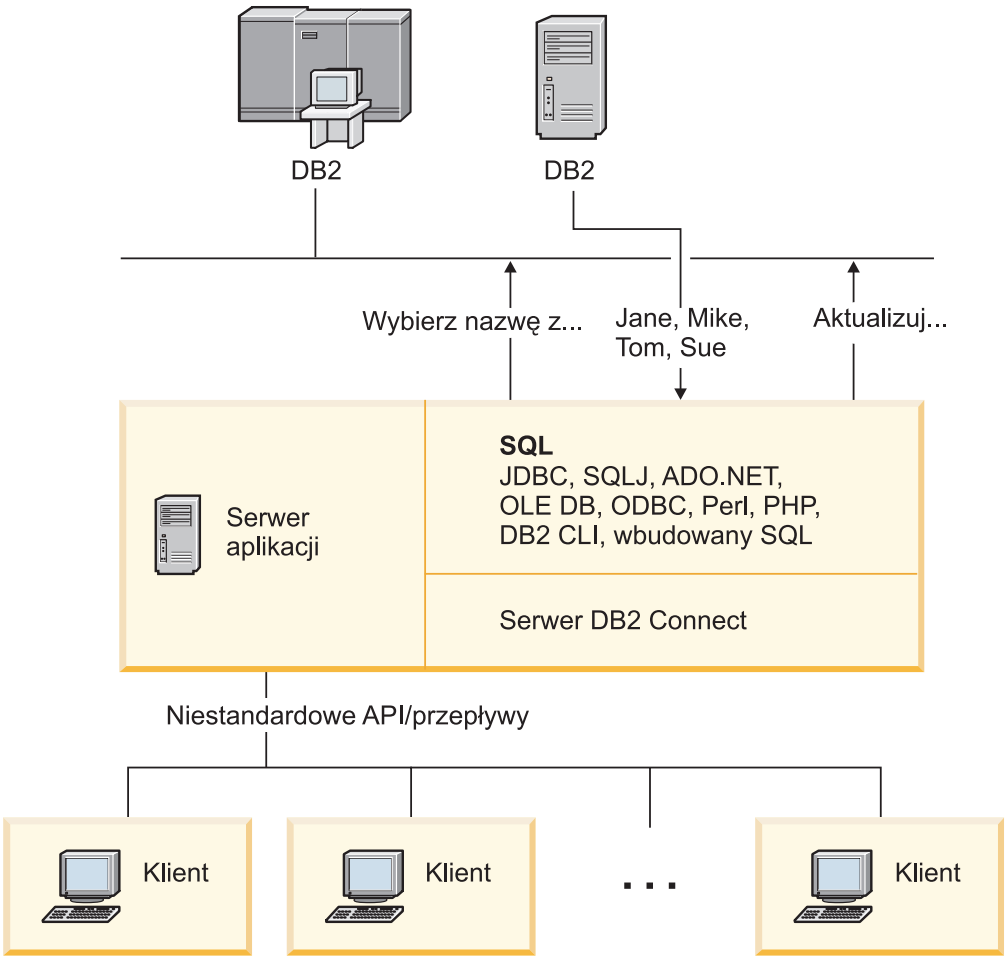
Produkty serwerowe DB2 Connect zapewniają wszechstronną pomoc przy wdrażaniu aplikacji wielowarstwowych. Funkcjonalność udostępniana przez produkt DB2 Connect obejmuje szeroki zakres interfejsów API, które mogą być używane do rozwijania logiki aplikacji (ODBC, ADO.NET, DB2 CLI, Embedded SQL, JDBC, SQLJ, Perl, PHP i OLE DB) oraz pełną infrastrukturę komunikacyjną do współdziałania z serwerami baz danych z rodziny DB2.

Produkt DB2 Connect wspomaga również te implementacje, w których warstwa bazy danych składa się z wielu serwerów baz danych z rodziny DB2. Umożliwia to serwerom aplikacji implementowanie transakcji, które w ramach jednej transakcji aktualizują dane znajdujące się na wielu serwerach baz danych.

Udostępniana przez produkt DB2 Connect obsługa protokołu zatwierdzania dwufazowego zapewnia zachowanie integralności takich rozproszonych transakcji. Na przykład aplikacja może aktualizować dane w bazach danych DB2 for OS/390 and z/OS oraz DB2 Database for Linux, UNIX, and Windows w ramach tej samej

transakcji. Jeśli jest zainstalowana i włączona obsługa żądań rozproszonych, aplikacja może w ramach jednej transakcji czytać z baz danych Oracle i aktualizować bazy danych rodziny DB2.

Na poniższym diagramie interfejsy API oraz mechanizm łączności pomiędzy serwerem aplikacji i serwerami baz danych zaplecza są realizowane za pośrednictwem produktu serwerowego DB2 Connect, takiego jak DB2 Connect Enterprise Edition.



Rysunek 6. Obsługa serwerów aplikacji w programie DB2 Connect

Zaawansowane opcje produktu DB2 Connect, takie jak zestawianie połączeń, znacznie zmniejszają wymagania aplikacji dotyczące zasobów oraz upraszczają implementację serwera aplikacji.

DB2 Connect i konfiguracje serwerów aplikacji

W przypadku serwerów aplikacji użycie produktu serwerowego DB2 Connect jest wymagane. Program DB2 Connect Personal Edition nie może być używany z serwerami aplikacji i nie istnieje na to licencja. Ponadto klienci implementujący serwery aplikacji powinni przeczytać warunki dołączone do kopii produktu DB2 Connect, aby dowiedzieć się, jaka jest wymagana liczba licencji dla użytkowników.

Istnieją dwie metody wdrażania produktu DB2 Connect w środowisku serwera aplikacji. Produkt serwerowy DB2 Connect można zainstalować na:

- serwerze aplikacji
- oddzielnym serwerze komunikacyjnym

W większości przypadków zalecanym rozwiązaniem jest zainstalowanie kopii produktu DB2 Connect na tym samym serwerze, na którym znajduje się serwer aplikacji. Zainstalowanie programu DB2 Connect na serwerze aplikacji pozwala mu brać udział we wszystkich schematach działań po wystąpieniu awarii oraz schematach równoważenia obciążenia, które może realizować serwer aplikacji. Taka instalacja może dać lepszą wydajność, ponieważ eliminuje dodatkowe przeskoki w sieci wymagane, jeśli program DB2 Connect zostanie zainstalowany na oddzielnym serwerze. Ponadto można dzięki temu uprościć administrację, ponieważ nie trzeba instalować i obsługiwać dodatkowego serwera.

Instalowanie produktu DB2 Connect na oddzielnym serwerze jest wygodne w sytuacjach, gdy produkt serwerowy DB2 Connect nie jest dostępny w wersji dla systemu operacyjnego lub platformy sprzętowej serwera aplikacji.

Program DB2 Connect i monitory przetwarzania transakcyjnego

Serwer aplikacji umożliwia dużej liczbie użytkowników wykonywanie aplikacji przy użyciu minimalnej ilości zasobów systemowych. Serwer aplikacji można rozszerzyć w taki sposób, aby pozwalał na wywoływanie skoordynowanych transakcji z aplikacji wykonywanych przez serwer aplikacji. Taka koordynacja transakcji jest powszechnie znana jako monitor przetwarzania transakcyjnego (TP, transaction processing). Monitor przetwarzania transakcyjnego działa w powiązaniu z serwerem aplikacji.

Transakcję można traktować jak procedurę zdarzenia (zazwyczaj jest to żądanie usługi) uruchomionego w ramach normalnych, codziennych działań wykonywanych w organizacji. Przetwarzanie transakcji w określonej kolejności to główne zadanie monitorów przetwarzania transakcyjnego.

Przetwarzanie transakcyjne

W każdej organizacji istnieją zasady i procedury opisujące sposób, w jaki organizacja ta powinna działać. Aplikacje użytkowników implementujące te zasady można nazwać *logiką biznesową*. Transakcje wykonywane przez takie aplikacje komercyjne często nazywane są przetwarzaniem transakcyjnym lub przetwarzaniem transakcyjnym otwartej bazy danych (OLPT, Online Transaction Processing).

Najważniejsze cechy komercyjnego przetwarzania OLTP:

Wielu użytkowników
Większość użytkowników w organizacji korzysta z przetwarzania transakcyjnego, ponieważ wiele osób ma wpływ na bieżący stan działań biznesowych.

Powtarzalność
Większość działań wykonywanych za pomocą komputera to zwykle ten sam proces, który jest wykonywany wielokrotnie. Na przykład wprowadzanie zamówień lub przetwarzanie płatności odbywa się wielokrotnie w ciągu dnia.

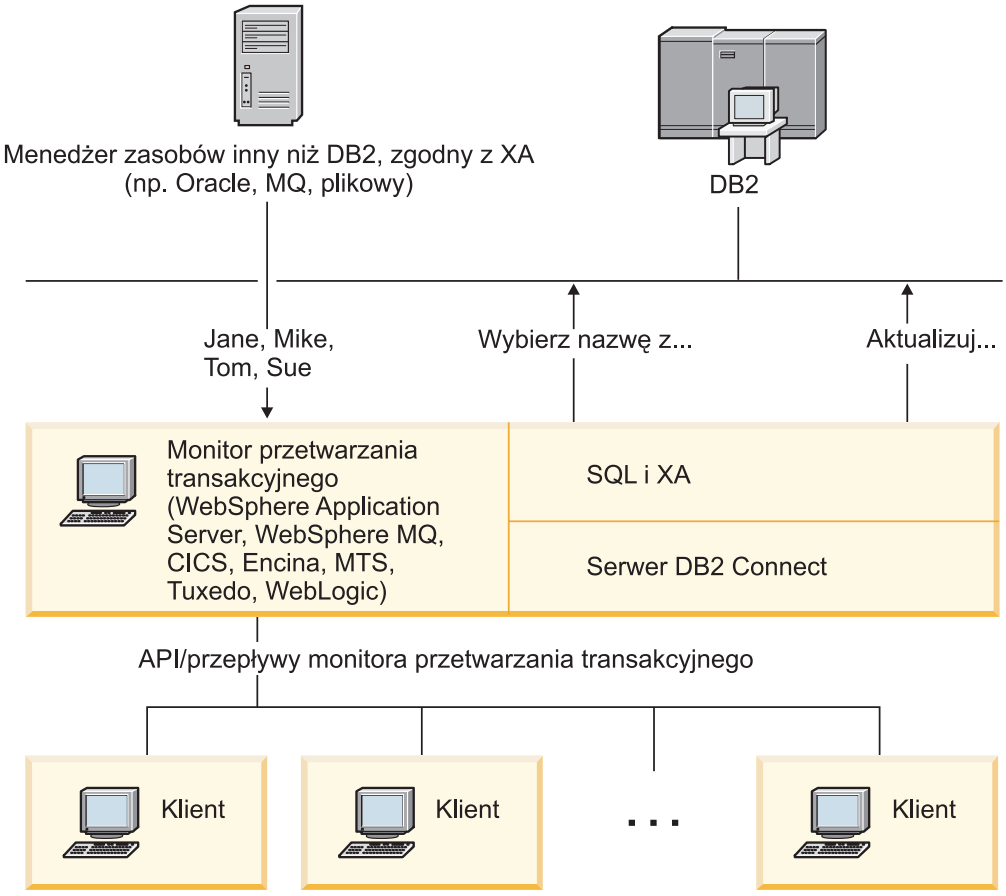
Krótkotrwałe operacje
Większość zadań wykonywanych przez pracowników za pomocą systemu przetwarzania transakcyjnego to operacje krótkotrwałe.

Współużytkowanie danych
Może istnieć tylko jedna kopia danych, ponieważ dane reprezentują stan organizacji.

Integralność
Dane muszą reprezentować bieżący stan organizacji, dlatego też muszą być wewnętrznie spójne. Na przykład każde zamówienie musi być związane z konkretnym rekordem klienta.

Niski koszt/transakcje

Koszt użytkownika systemu powinien być minimalny, ponieważ przetwarzanie transakcyjne generuje bezpośredni koszt funkcjonowania firmy. Produkt DB2 Connect umożliwia aplikacjom z serwera aplikacji działającego pod kontrolą systemu operacyjnego Linux, UNIX lub Windows wykonywanie transakcji w zdalnej sieci LAN, na zdalnym hoście i na zdalnych serwerach bazy danych System i oraz koordynowanie ich przy użyciu monitora przetwarzania transakcyjnego.



Rysunek 7. Obsługa monitorów przetwarzania transakcyjnego w produkcie DB2 Connect

Na rysunku Rys. 7 interfejsy API, podobnie jak mechanizm połączeń między serwerem aplikacji i serwerami baz danych zaplecza, są udostępniane za pośrednictwem produktu serwerowego DB2 Connect, takiego jak DB2 Connect Enterprise Edition.

Przykłady monitorów przetwarzania transakcyjnego

Do najbardziej popularnych monitorów przetwarzania transakcyjnego dostępnych obecnie na rynku należą:

- IBM WebSphere Application Server
- IBM WebSphere MQ
- IBM TxSeries CICS
- BEA Tuxedo
- BEA WebLogic
- Microsoft Transaction Server (MTS)

W ramach transakcji koordynowanych przez te monitory przetwarzania transakcyjnego mogą być używane zdalne serwery baz danych na serwerach System i, zSeries i w sieciach LAN.

Model rozproszonego przetwarzania transakcyjnego (DTP) X/Open

Do aktualizacji wielu zasobów w ramach jednej transakcji może być wymagana aplikacja wykonująca logikę biznesową. Na przykład aplikacja bankowa, której zadaniem jest przekazywanie pieniędzy z jednego konta na inne, może wymagać debetowania jednej bazy danych (rachunek "od") i kredytowania drugiej bazy (rachunek "dla").

Dwie bazy danych mogą również pochodzić od różnych dostawców. Na przykład jedną bazą danych może być DB2 Universal Database for OS/390 and z/OS a drugą baza danych Oracle. Zamiast monitorów, z których każdy implementuje interfejs transakcji prawnie zastrzeżony dla dostawcy bazy danych, zdefiniowany został jeden wspólny interfejs transakcji między monitorem przetwarzania transakcyjnego i wszystkimi zasobami, z których korzysta aplikacja. Interfejs ten jest znany pod nazwą *interfejsu XA*. Monitor TP, który używa interfejsu XA, nazywany jest *menedżerem transakcji (TM, Transaction Manager) zgodnym z interfejsem XA*. Zasób dający się aktualizować, w którym został zaimplementowany interfejs XA, nazywamy *menedżerem zasobów (RM, Resource Manager) zgodnym z XA*.

Wszystkie monitory transakcji wymienione powyżej są zgodne z interfejsem XA. Bazy danych zdalnego hosta, bazy danych System i i oparte na sieci LAN bazy danych DB2 podczas udostępniania danych produktowi DB2 Connect są modułami żądań zgodnymi z interfejsem XA. Dlatego każdy monitor przetwarzania transakcyjnego dysponujący zgodnym z interfejsem XA menedżerem TM może wewnątrz aplikacji biznesowych wykonujących transakcje korzystać z baz danych hosta, baz danych System i oraz baz danych DB2 opartych na sieci LAN.

Część 2. Informacje dodatkowe o produkcie DB2 Connect

© Copyright IBM Corp. 1993, 2010

Rozdział 4. Aktualizowanie katalogów bazy danych

Do zarządzania informacjami o połączeniach z bazą danych program DB2 Connect wykorzystuje następujące katalogi:

- *katalog systemowy bazy danych*, który zawiera informacje dotyczące nazw węzłów i uwierzytelniania dla wszystkich baz danych, z którymi łączy się program DB2 Connect.
- *katalog węzłów*, który zawiera informacje o adresach sieciowych i protokołach komunikacyjnych wszystkich hostów i serwerów baz danych System i, z którymi łączy się program DB2 Connect.
- *katalog dostępu do bazy danych (DCS)*, który zawiera informacje dotyczące baz danych serwerów na hostach lub serwerach System i.

Uwaga:

1. Przed aktualizacją tych katalogów należy skonfigurować komunikację między serwerem bazy danych na hoście lub serwerze System i a stacjami roboczymi.
2. Katalogi bazy danych można zaktualizować za pomocą Asysty podczas konfigurowania.

Aby zaktualizować katalogi bazy danych:

1. Zbierz informacje o katalogu bazy danych przy użyciu arkusza dostosowania katalogu
2. Przeczytaj temat “Aktualizacja katalogów informacjami o zdalnych serwerach baz danych” w Centrum informacyjnym

Wartości w systemowym katalogu bazy danych

Systemowy katalog bazy danych jest tworzony dla każdej instancji menedżera bazy danych i zawiera jedną pozycję dla każdej bazy danych, która została wpisana do katalogu dla tej instancji. W produktach DB2 Connect systemowy katalog bazy danych zawiera nazwę, alias, nazwę węzła i typ uwierzytelniania dla każdej bazy danych.

W systemowym katalogu baz danych można określić następujące informacje:

Nazwa bazy danych
Ta sama wartość, którą podano w tabeli parametrów katalogu DCS.

Alias bazy danych
Alias serwera bazy danych na serwerze zSeries lub System i. Nazwa ta będzie używana przez aplikację, która korzysta z bazy danych. Wartością domyślną jest nazwa określona dla bazy danych.

Format: 1-8 jednobajtowych znaków alfanumerycznych łącznie ze znakiem (#), (@), ($) i znakiem podkreślenia (_). Nazwa nie może zaczynać się znakiem podkreślenia ani cyfrą.

Nazwa węzła
Ta sama wartość, którą podano w tabeli parametrów katalogu węzłów.

Uwierzytelnianie
Określa miejsce, w którym nastąpi sprawdzenie poprawności nazwy użytkownika i hasła dla połączeń pochodzących z serwera DB2 Connect. Poprawne opcje to: SERVER, SERVER_ENCRYPT, CLIENT, KERBEROS, SERVER_ENCRYPT_AES i DATA_ENCRYPT. Typ uwierzytelniania GSSPLUGIN nie jest obsługiwany w systemowym katalogu bazy danych.

© Copyright IBM Corp. 1993, 2010

Wartości w katalogu węzłów

W katalogu węzłów można wpisać następujące informacje:

Nazwa węzła
Pseudonim systemu serwera bazy danych na hoście lub serwerze System i, w którym rezyduje zdalna baza danych. Tę nazwę definiuje użytkownik. Należy ją wpisać w tabeli parametrów katalogu węzłów i w tabeli parametrów systemowego katalogu bazy danych.

Format: 1-8 jednobajtowych znaków alfanumerycznych łącznie ze znakiem (#), (@), ($) i znakiem podkreślenia (_). Nazwa nie może zaczynać się znakiem podkreślenia ani cyfrą.

Protokół
Obowiązkowy jest protokół TCP/IP.

Typ zabezpieczeń
Używany typ zabezpieczeń. W przypadku węzłów TCP/IP opcja SECURITY SOCKS określa, że węzeł będzie korzystał z mechanizmu SOCKS. Jeśli wybrano tę opcję, zmienne środowiskowe SOCKS_NS i SOCKS_SERVER są obowiązkowe i muszą być ustawione tak, aby mechanizm SOCKS był dostępny.

Nazwa TCP/IP zdalnego hosta lub adres IP
W przypadku węzła TCP/IP jest to nazwa hosta TCP/IP lub adres TCP/IP. Jeśli podano nazwę hosta, musi ona zostać przekształcona na stacji roboczej DB2 Connect za pomocą serwera DNS (Domain Name Server) lub na podstawie lokalnego pliku hostów TCP/IP.

W przypadku zdalnych hostów DB2 for OS/390 and z/OS nazwa hosta jest wyświetlana w komunikacie DSNL004I (DOMAIN=nazwa hosta) podczas uruchamiania narzędzia obsługi danych rozproszonych (Distributed Data Facility - DDF). Można także użyć komendy -DISplay DDF.

Podczas uzyskiwania dostępu do grupy współużytkującej dane w systemie z/OS nazwa domenowa powinna być odwzorowana na dynamiczny adres VIPA grupy DB2. Adres ten umożliwia przekierowanie do najmniej obciążonego elementu programu DB2. Aby uzyskać dostęp do konkretnego elementu, należy użyć dynamicznego adresu VIPA tego elementu DB2 i wyłączyć przekierowania sysplex. W każdym komunikacie DSNL004I dotyczącym określonego elementu dostępna jest nazwa domenowa dla danego elementu.

Nazwa usługi TCP/IP lub numer portu
W przypadku węzła TCP/IP jest to zdalna nazwa usługi TCP/IP lub numer portu. Musi ona być zdefiniowana na zdalnym hoście TCP/IP. Numer portu 446 został zarejestrowany jako domyślny numer portu dla architektury DRDA.

W przypadku zdalnych hostów DB2 for OS/390 and z/OS numer portu jest zdefiniowany w zestawie danych programu startowego (Boot Strap Data Set - BSDS) jako PORT i jest wyświetlany w komunikacie DSNL004I (TCPPORT=numer portu) podczas uruchamiania narzędzia DDF. Można także użyć komendy -DISplay DDF.

Podczas uzyskiwania dostępu do grupy współużytkującej dane w systemie z/OS nazwa domenowa powinna być odwzorowana na dynamiczny adres VIPA grupy DB2. Adres ten umożliwia przekierowanie do najmniej obciążonego elementu programu DB2. Aby uzyskać dostęp do konkretnego elementu, należy użyć dynamicznego adresu VIPA tego elementu DB2 i wyłączyć przekierowania sysplex. W każdym komunikacie DSNL004I dotyczącym określonego elementu dostępna jest nazwa domenowa dla danego elementu.

Uwaga: Serwer może przypisać drugi port używany przy operacjach ponownego synchronizowania zatwierdzania dwufazowego w połączeniach TCP/IP. Przykładowo zestaw danych programu startowego DB2 Universal Database for z/OS and OS/390 przypisuje numer portu (RESPORT), który ma być używany tylko do ponownego synchronizowania połączeń przychodzących do produktu DB2 Universal Database for z/OS and OS/390. Nie należy podawać tu nazwy usługi.

Wartości w katalogu DCS

W katalogu DCS można umieścić następujące informacje:

Nazwa bazy danych
Zdefiniowany przez użytkownika pseudonim serwera bazy danych na hoście lub serwerze System i. Należy go wpisać w tabeli parametrów katalogu DCS i w tabeli parametrów systemowego katalogu baz danych.

Format: 1-8 jednobajtowych znaków alfanumerycznych łącznie ze znakiem (#), (@), ($) i znakiem podkreślenia (_). Nazwa nie może zaczynać się znakiem podkreślenia ani cyfrą.

Nazwa docelowej bazy danych
Jedna z poniższych baz danych w systemie serwera bazy danych na hoście lub serwerze System i:

OS/390 i z/OS
Podsystem DB2 Universal Database for z/OS and OS/390 identyfikowany przez parametr LOCATION NAME lub jedną ze zdefiniowanych na serwerze z/OS nazw aliasów LOCATION.

LOCATION NAME można określić, logując się do TSO i wysyłając następujące zapytanie SQL za pomocą jednego z dostępnych narzędzi tworzenia zapytań:

```
select current server from sysibm.sysdummy1
```

Definicje wielu wartości parametru LOCATION NAME znajdują się także w zestawie danych programu startowego (BSDS, Boot Strap Data Set), jak również w komunikacie DSNL004I (LOCATION=położenie), który jest zapisywany w momencie uruchamiania programu Distributed Data Facility (DDF). Można także użyć komendy -DISplay DDF.

Podczas uzyskiwania dostępu do grupy współużytkującej dane w systemie z/OS nazwa domenowa powinna być odwzorowana na dynamiczny adres VIPA grupy DB2. Adres ten umożliwia przekierowanie do najmniej obciążonego elementu programu DB2. Aby uzyskać dostęp do konkretnego elementu, należy użyć dynamicznego adresu VIPA tego elementu DB2 i wyłączyć przekierowania sysplex. W każdym komunikacie DSNL004I dotyczącym określonego elementu dostępna jest nazwa domenowa dla danego elementu.

VSE lub VM
Nazwa bazy danych (DBNAME).

OS/400 i z/OS
Nazwa relacyjnej bazy danych (RDBNAME).

Inne W przypadku systemów operacyjnych Windows, Linux i UNIX alias bazy danych znaleziony w katalogu bazy danych.

Łańcuch parametrów
Aby zmienić wartości domyślne, należy podać jeden lub wszystkie poniższe parametry w następującej kolejności.

plik-odwzorowania

Nazwa pliku odwzorowania SQLCODE, który nadpisuje domyślne odwzorowanie SQLCODE. Aby wyłączyć odwzorowanie SQLCODE, należy podać wartość **NOMAP**.

Uwaga: Podczas przetwarzania żądania zapytania serwer DRDA zwraca dane w postaci zestawu wierszy reprezentujących tabelę wynikową. Z każdym wierszem zwracany jest także obszar SQLCA zawierający zwykle zerowe lub dodatnie wartości kodu sqlcode (na przykład +12 lub +802). Jeśli na serwerze DB2 Connect używany jest niestandardowy plik odwzorowań, takie dodatnie wartości kodów sqlcode nie będą odwzorowane, o ile znajdują się w tym pliku i mają niestandardowe odwzorowania (np. są odwzorowywane na inne kody sqlcode lub mają niestandardowe odwzorowania elementów zmiennych).

Należy podkreślić, że:

1. Dodatnie wartości kodów sqlcode oznaczają ostrzeżenia, w przeciwieństwie do kodów ujemnych, które wskazują na błędy. Wszystkie ujemne wartości kodów sqlcode będą zawsze odwzorowywane, niezależnie od okoliczności oraz od używanego pliku odwzorowań. Wszystkie dodatnie kody sqlcode zawarte w niestandardowym pliku odwzorowań są zawsze odwzorowywane na siebie bez żadnych zmian. Także te dodatnie kody sqlcode, których nie ma w niestandardowym pliku odwzorowań na serwerze DB2 Connect, są zawsze odwzorowywane.
2. Jeśli używany jest domyślny plik odwzorowań lub istnieje bezpośrednie połączenie z bazą danych na hoście, odwzorowanie jest zawsze przeprowadzane dla wszystkich kodów sqlcode.

,D Jest to parametr zajmujący drugą pozycję. Jeśli zostanie on określony, aplikacja rozłączy się z serwerem bazy danych na hoście lub serwerze System i po zwróceniu jednego z następujących kodów SQLCODE:

```
SQL30000N
SQL30040N
SQL30050N
SQL30051N
SQL30053N
SQL30060N
SQL30070N
SQL30071N
SQL30072N
SQL30073N
SQL30074N
SQL30090N
```

Jeśli nie podano parametru rozłączenia (**,D**), rozłączenie nastąpi tylko po zwróceniu następujących kodów SQLCODE:

```
SQL30020N
SQL30021N
SQL30041N
SQL30061N
SQL30081N
```

Wyjaśnienia dotyczące kodów można znaleźć w podręczniku *Message Reference*.

Uwaga: Jeśli program DB2 Connect rozłącza się z powodu błędu, wycofanie zmian następuje automatycznie.

„INTERRUPT_ENABLED

Jest to parametr zajmujący trzecią pozycję. Parametr INTERRUPT_ENABLED ma zastosowanie tylko wtedy, gdy serwer końcowy nie obsługuje przerwań. Jeśli serwer obsługuje związane z przerwaniami przepływy danych w architekturze DRDA, produkt DB2 Connect po prostu przekaże żądanie przerwania do serwera.

Jeśli parametr INTERRUPT_ENABLED zostanie skonfigurowany w katalogu DCS na stacji roboczej DB2 Connect, a aplikacja klienta zgłosi przerwanie podczas połączenia z serwerem bazy danych na hoście lub serwerze System i, program DB2 Connect wykona przerwanie, kończąc połączenie i wycofując zmiany w jednostce pracy. Takie zachowanie związane z przerwaniem jest obsługiwane w systemach AIX i Windows.

Aplikacja otrzymuje kod sqlcode (-30081) wskazujący, że połączenie z serwerem zostało zakończone. W celu przetworzenia dalszych żądań do bazy danych aplikacja musi nawiązać nowe połączenie z serwerem bazy danych na hoście lub serwerze System i. Na platformach innych niż AIX w wersji 5.2 lub późniejszej oraz Windows, produkt DB2 Connect nie obsługuje opcji automatycznego rozłączenia wykonywanego w chwili otrzymania przez aplikację żądania przerwania.

Uwaga: Opcja ta jest obsługiwana na wszystkich platformach w przypadku połączeń TCP/IP. Klient może zabić gniazdo, ale - w zależności od implementacji serwera - może, ale nie musi wystąpić wyjątkowy odbiór. Program DB2 Universal Database for z/OS and OS/390 używa asynchronicznych wywołań gniazd, a zatem może wykryć utratę połączenia i wycofać wszelkie uruchomione instrukcje SQL.

„„„„SYSPLEX

Parametru na szóstej pozycji można użyć, aby w sposób jawny włączyć obsługę SYSPLEX DB2 Connect dla określonej bazy danych.

„„„„„LOCALDATE="<wartość>"

Ten parametr, zajmujący siódmą pozycję, jest stosowany do włączania obsługi formatowania daty przez produkt DB2 Connect. Zostało to zaimplementowane przy użyciu maski daty dla wartości <*wartość*> w następujący sposób:

Załóżmy, że uruchomiono następujące instrukcje procesora wiersza komend:

```
catalog TCPIP node nynode remote myhost server myport
catalog dcs database nydb1 as new_york
catalog database nydb1 as newyork1 at node nynode
     authentication server
```

Alias bazy danych *newyork1* ma być użyty do uzyskania dostępu do bazy danych hosta bez przekształcania daty, ponieważ nie określono żadnej maski daty.

Nowa obsługa formatowania daty pozwala jednak używać poniższych komend procesora wiersza komend. W tym przypadku, jako że używany jest procesor wiersza komend, a łańcuch parametrów ujęto w znaki podwójnego cudzysłowu, wartość LOCALDATE musi zostać ujęta w dwie pary podwójnych cudzysłowów. Warto zwrócić uwagę na użycie znaku zmiany znaczenia z systemu operacyjnego "\" (ukośnik odwrotny), który zapewnia, że znaki podwójnego cudzysłowu nie zostaną usunięte ze specyfikacji LOCALDATE.

```
catalog dcs database nydb2 as new_york
     parms \",,,,,,LOCALDATE=\"\"YYYYMMDD\"\"\"
catalog database nydb2 as newyork2 at node nynode
     authentication server
```

Alias bazy danych newyork2 daje dostęp do tej samej bazy danych na hoście, a ponadto określono dla niego maskę formatu daty. Przykład ten pokazuje, że maska daty jest określana za pomocą parametru LOCALDATE i stanowi ona parametr zajmujący siódmą pozycję w polu PARMS pozycji katalogu DCS.

Aby maska daty była poprawna, muszą zostać spełnione WSZYSTKIE następujące warunki:

1. Może istnieć tylko jeden ciąg znaków Y, M i D, przy czym Y oznacza cyfrę roku, M - cyfrę miesiąca, a D - cyfrę dnia.
2. Maksymalna liczba znaków Y w ciągu wynosi 4.
3. Maksymalna liczba znaków M w ciągu wynosi 2.
4. Maksymalna liczba znaków D w ciągu wynosi 2.

Następujące maski daty są poprawne:

```
"YYyyMmDd"   - znaki Y, M i D mogą być małe lub wielkie
"MM+DD+YYYY" - maska może być dłuższa niż 10 bajtów
               i może zawierać znaki inne niż Y, M
               i D
"abcYY+MM"   - ciąg znaków D nie musi wystąpić
```

Następujące maski daty są niepoprawne:

```
"YYYYyMMDD"  - niepoprawny jest piąty znak Y w sekwencji
"YYYYMDDM"   - niepoprawne są dwie sekwencje znaków M
```

Jeśli maska daty jest niepoprawna, nie wystąpi błąd. Zostanie ona po prostu zignorowana. To, że maska daty jest poprawna, nie oznacza jeszcze, że będzie używana. Przekształcenie formatu daty oparte na poprawnej masce zostanie wykonane, jeśli spełnione będą WSZYSTKIE następujące warunki:

1. Nie wystąpi błąd SQL.
2. Wartością wyjściową jest data w formacie ISO lub JIS.
3. Obszar wyjściowy daty ma długość przynajmniej 10 bajtów. Jest to minimalna wielkość obszaru wyjściowego daty, nawet jeśli nie będzie wykonywane ŻADNE przekształcanie formatu daty. Warunek ten musi być spełniony, nawet gdy maska daty jest krótsza niż 10 bajtów.

4. W pozycji katalogu DCS określono poprawną maskę daty i pasuje ona do obszaru wyjściowego daty.

,,,,,,,,BIDI=<ccsid>

Dziewiąty parametr pozycyjny jest używany do określania dwukierunkowego (BiDi) identyfikatora CCSID, który ma być użyty do zastąpienia domyślnego dwukierunkowego identyfikatora CCSID bazy danych serwera. Na przykład:

```
",,,,,,,,BIDI=xyz"
```

gdzie *xyz* reprezentuje zastąpienie identyfikatora CCSID.

Arkusz dostosowania katalogu

Arkusz dostosowania katalogu zawiera informacje, które muszą zostać zebrane. Dla własnej wygody można skopiować ten arkusz i wpisać w nim wartości używanego systemu.

Parametry katalogu węzłów.

Tabela 1. Parametry katalogu węzłów.

Parametr	Przykład	Wartość użytkownika
Nazwa węzła	DB2NODE	
Nazwa hosta zdalnego (węzeł TCP/IP)	ZOSHOST	
Serwer (nazwa usługi TCP/IP lub numer portu)	db2inst1c (lub 446)	

Uwaga:

1. Domyślny numer portu TCP/IP dla DRDA wynosi 446.
2. Jeśli nie wiadomo, czy serwer bazy danych na hoście lub serwerze System i obsługuje opcję SECURITY SOCKS, nie należy określać parametru SECURITY dla węzła TCP/IP.

Parametry katalogu DCS.

Tabela 2. Parametry katalogu DCS.

Parametr	Przykład	Wartość użytkownika
Nazwa bazy danych	DB2DB	
Nazwa docelowej bazy danych	NEW_YORK3	
Requester aplikacji		
Łańcuch parametrów	",,,,,,LOCALDATE=\"\"YYMMDD\"\"\"	

Parametry systemowego katalogu baz danych.

Tabela 3. Parametry systemowego katalogu baz danych.

Parametr	Przykład	Wartość użytkownika
Nazwa bazy danych	DB2DB	
Alias bazy danych	NYC3	
Nazwa węzła	DB2NODE	
Uwierzytelnianie	SERVER	

Definiowanie wielu pozycji dla tej samej bazy danych

Dla każdej bazy danych należy zdefiniować przynajmniej jedną pozycję w każdym z trzech katalogów (katalogu węzłów, katalogu DCS i systemowym katalogu baz danych). W pewnych przypadkach może być konieczne zdefiniowanie kilku pozycji dla jednej bazy danych.

Może na przykład wystąpić konieczność wyłączenia odwzorowania kodów SQLCODE dla aplikacji przeniesionych z serwera bazy danych na hoście lub serwerze System i, które akceptują domyślne odwzorowania dla aplikacji napisanych dla środowisk typu klient/serwer. Można to zrobić w następujący sposób:

- Zdefiniować jedną pozycję w katalogu węzłów.
- Zdefiniować dwie pozycje w katalogu DCS z różnymi nazwami bazy danych. W jednej pozycji należy w łańcuchu parametrów podać wartość NOMAP.
- Zdefiniować dwie pozycje w systemowym katalogu baz danych z różnymi aliasami bazy danych z dwiema nazwami bazy danych określonymi w katalogu DCS.

Oba aliasy odwołują się do tej samej bazy danych: jeden używa przy tym odwzorowania SQLCODE, a drugi nie.

Obsługa danych dwukierunkowych

Poniższa sekcja dotyczy wyłącznie serwerów OS/390 i z/OS. Nie należy włączać tej opcji w wypadku serwera DB2 for i5/OS, ponieważ serwer ten zapewnia pełną obsługę danych dwukierunkowych.

Następujące atrybuty dwukierunkowe (BiDi) są wymagane do poprawnej obsługi danych dwukierunkowych na różnych platformach:

- kształt numeryczny (ARABIC lub HINDI)
- orientacja (RIGHT-TO-LEFT lub LEFT-TO-RIGHT)
- kształtowanie (SHAPED lub UNSHAPED)
- wymiana symetryczna (YES lub NO)
- typ tekstu (LOGICAL lub VISUAL)

Ponieważ wartości domyślne na różnych platformach nie są takie same, przy przesyłaniu danych DB2 z jednej platformy na drugą mogą wystąpić problemy. Na przykład platformy Windows korzystają z danych w formacie LOGICAL UNSHAPED, podczas gdy dane z platform OS/390 lub z/OS występują zwykle w formacie SHAPED VISUAL. Dlatego bez wsparcia atrybutów dwukierunkowych, dane wysłane z produktu DB2 for OS/390 and z/OS do produktu DB2 Connect w systemie Windows wyświetlane są nieprawidłowo.

Gdy dane są wymienianie między programem DB2 Connect a bazą danych znajdującą się na serwerze, zazwyczaj odbiorca wykonuje konwersję danych przychodzących. Ta sama zasada ma również zastosowanie do dwukierunkowej transformacji układu, występującej oprócz zwykłej konwersji strony kodowej. Jednak obecnie żaden produkt DB2 dla hostów nie obsługuje identyfikatorów CCSID dla znaków dwukierunkowych ani transformacji układu BiDi. Dlatego też ulepszono program DB2 Connect, wprowadzając możliwość wykonywania dwukierunkowej transformacji układu dla danych przesyłanych do bazy danych serwera (oprócz możliwości wykonania tej transformacji dla danych odebranych z bazy danych serwera).

Aby produkt DB2 Connect wykonał dwukierunkową transformację układu danych wychodzących do bazy danych serwera, przesłonięty musi zostać identyfikator BiDi CCSID bazy danych serwera. Można to wykonać, używając parametru BIDI w polu PARMS pozycji katalogu DCS dla bazy danych serwera.

Użycie tej opcji przedstawia następujący przykład.

Rozważmy klienta IBM Data Server Client z językiem hebrajskim działającego z identyfikatorem CCSID 62213 (łańcuch BiDi typu 5), który ma łączyć się z bazą danych DB2 na hoście działającą z identyfikatorem CCSID 424 (łańcuch BiDi typu 4). Wiadomo przy tym, że dane zawarte w bazie danych DB2 na hoście wykorzystują identyfikator CCSID 62245 (łańcuch BiDi typu 10).

W takiej sytuacji występują dwa problemy. Po pierwsze, w bazie danych DB2 na hoście nie jest znana różnica między typami łańcuchów BiDi o identyfikatorach CCSID 424 i 62245. Po drugie, w bazie danych hosta DB2 nie jest rozpoznawany identyfikator CCSID 62213 klienta IBM Data Server Client. W bazie tej obsługiwany jest tylko identyfikator CCSID 62209 (łańcuch BiDi typu 10), który bazuje na tej samej stronie kodowej, na której bazuje identyfikator CCSID 62213.

Należy upewnić się, że dane wysłane do bazy danych hosta DB2 są w formacie łańcucha BiDi typu 6, aby poinformować produkt DB2 Connect o konieczności wykonania dwukierunkowej transformacji układu dla danych odbieranych z bazy danych hosta DB2. Dla bazy danych DB2 na hoście konieczne jest zastosowanie następującego wpisu do katalogu:

```
catalog dcs database nydb1 as TELAVIV parms ",,,,,,,BIDI=62245"
```

Ten wpis informuje produkt DB2 Connect o konieczności nadpisania identyfikatora CCSID bazy danych hosta DB2 o dotychczasowej wartości 424 nową wartością 62245. Nadpisanie to obejmuje następujące przetwarzanie:

1. Produkt DB2 Connect połączy się z bazą danych hosta DB2, używając identyfikatora CCSID 62209 (łańcuch BiDi typu 10).
2. Produkt DB2 Connect wykona dwukierunkową transformację układu dla danych wysyłanych do bazy danych hosta DB2 z identyfikatora CCSID 62213 (łańcuch BiDi typu 5) na identyfikator CCSID 62209 (łańcuch BiDi typu 10).
3. Produkt DB2 Connect wykona dwukierunkową transformację układu dla danych odebranych z bazy danych hosta DB2 z identyfikatora CCSID 62245 (łańcuch BiDi typu 10) na identyfikator CCSID 62213 (łańcuch BiDi typu 5).

Uwaga:

1. Aby parametr BIDI został uwzględniony, należy ustawić dla zmiennej środowiskowej lub rejestru **DB2BIDI** wartość YES. Wartość zmiennej **DB2BIDI** należy określić na stacji roboczej DB2 Connect, na której wpisano do katalogu pozycję bazy danych DCS. W przypadku aplikacji uruchamianych na kliencie korzystającym ze zdalnego serwera DB2 Connect, zmienna **DB2BIDI** musi zostać ustawiona również na kliencie.
2. Aby program DB2 Connect wykonywał transformację układu dla danych, które ma wysłać do bazy danych DB2 na hoście, nawet wtedy, gdy nie ma konieczności nadpisywania identyfikatora CCSID, trzeba dodać parametru BIDI w polu PARMS katalogu bazy danych DCS. W takim przypadku jako identyfikator CCSID należy podać domyślny identyfikator CCSID bazy danych DB2 na hoście.
3. W niektórych przypadkach użycie dwukierunkowego identyfikatora CCSID może spowodować taką modyfikację zapytania SQL, że stanie się ono nierozpoznawalne dla serwera DB2. Szczególnie należy unikać używania identyfikatorów CCSID IMPLICIT CONTEXTUAL i IMPLICIT RIGHT-TO-LEFT, gdy można użyć innego typu łańcucha. Działanie identyfikatorów CCSID CONTEXTUAL może być nieprzewidywalne, jeśli

zapytanie SQL zawiera łańcuchy w cudzysłowie. Należy unikać używania łańcuchów z cudzysłowami w instrukcjach SQL i w miarę możliwości zamiast nich używać zmiennych języka macierzystego.

Jeśli określony dwukierunkowy identyfikator CCSID powoduje problemy, których nie można rozwiązać, stosując uwzględniając powyższe zalecenia, należy ustawić dla zmiennej środowiskowej lub rejestru **DB2BIDI** wartość NO.

Specyfikacje łańcucha parametrów

Poniżej podano przykłady parametrów DCS (każdy wiersz to zestaw parametrów):

```
NOMAP
/u/username/sqllib/map/dcs1new.map,D
,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE="YYMMDD",,
```

Można też zaakceptować wartości domyślne, nie podając łańcucha parametrów.

Uwaga: Ze względu na konieczność podania dwóch par podwójnych znaków cudzysłowu przy określaniu maski LOCALDATE w łańcuchu parametrów należy zastosować systemowy znak zmiany znaczenia "\" (ukośnik odwrotny) przy korzystaniu z procesora CLP z poziomu wiersza komend systemu operacyjnego UNIX. Na przykład:

```
db2 catalog dcs db x as y parms \",,,,,,LOCALDATE=
            \"\"YYMMDD\"\"\"
```

Wynikiem tej komendy jest następująca pozycja katalogu DCS:

```
Pozycja 1 DCS:

 Nazwa lokalnej bazy danych               = X
 Nazwa docelowej bazy danych              = Y
 Nazwa requestera aplikacji               =
 Parametry DCS                            = ,,,,,,LOCALDATE="YYMMDD"
 Komentarz                                =
 Poziom wersji katalogu DCS               = 0x0100
```

Rozdział 5. Bezpieczeństwo w programie DB2 Connect

Uwierzytelnianie użytkowników jest bardzo ważne w programie DB2 Connect, ponieważ może on być używany zarówno przez użytkowników lokalnych, jak i zdalnych. Omówiono połączenia zaufane i obsługę protokołu Kerberos, przedstawiono także rozważania dotyczące bezpieczeństwa baz danych hosta.

Zaufane połączenia za pośrednictwem produktu DB2 Connect

Niektóre serwery baz danych DB2 obsługują zaufane konteksty. *Zaufany kontekst* umożliwia administratorowi bazy danych między innymi zdefiniowanie warunków, po których spełnieniu aplikacja kliencka będzie mogła utworzyć zaufane połączenie. *Zaufane połączenie* może wykonywać czynności niedostępne dla zwykłego połączenia.

Istnieją dwa rodzaje zaufanych połączeń: niejawne i jawne. To, czy podczas tworzenia połączenia uzyska się jawne zaufane, niejawne zaufane czy też zwykłe połączenie, zależy od tego czy wysłano żądanie zaufanego połączenia i czy połączenie spełnia kryteria zdefiniowane w zaufanym kontekście na serwerze, zgodnie z podsumowaniem w tabeli Tabela 4.

Tabela 4. Typy połączeń będące wynikiem różnych kombinacji działań

	Połączenie spełnia kryteria serwera dla zaufanego połączenia	**Połączenie nie spełnia kryteriów serwera dla zaufanego połączenia**
Zażądano, aby połączenie było zaufanym połączeniem	Jawne zaufane połączenie	Zwykłe połączenie i ostrzeżenie SQL20360W (SQLSTATE 01679)
Nie zażądano, aby połączenie było zaufanym połączeniem	Niejawne zaufane połączenie	Zwykłe połączenie

Niejawne zaufane połączenie jest identyczne ze zwykłym połączeniem za wyjątkiem tego, że nadaje użytkownikowi tymczasowe uprawnienia roli na czas korzystania z połączenia. Nadane uprawnienia roli (jeśli występują) są określone w zaufanym kontekście, który spowodował, że połączenie jest zaufane.

Niejawne zaufane połączenia mogą być tworzone przez dowolną aplikację, która łączy się przy użyciu produktu DB2 Connect. Niejawne zaufane połączenia są tworzone i używane w taki sam sposób, w jaki tworzone i używane są zwykłe połączenia. Oznacza to, że nie są wymagane żadne zmiany w kodzie istniejących aplikacji, aby korzystać z niejawnych zaufanych połączeń, dopóki aplikacja łączy się poprzez produkt DB2 Connect.

Jawne zaufane połączenie nadaje użytkownikowi tymczasowe uprawnienia roli w ten sam sposób, co niejawne zaufane połączenie. Dodatkowo jawne zaufane połączenie pozwala na zmianę ID autoryzowanego użytkownika podczas wykonywania działań w trwającym połączeniu. Zmiana ID autoryzowanego użytkownika w jawnym zaufanym połączeniu jest nazywana *przełączaniem użytkowników*. ID autoryzowanego użytkownika, na który można się przełączyć, oraz to, czy przy przełączaniu do niego wymagane jest podanie hasła, zdefiniowane jest jako część zaufanego kontekstu użytego do utworzenie zaufanego połączenia.

Przełączanie użytkownika może znacząco zmniejszyć narzut związany ze współużytkowaniem połączenia przez kilku użytkowników, zwłaszcza dla nazw

© Copyright IBM Corp. 1993, 2010

użytkowników, które nie wymagają podania hasła, ponieważ w takim przypadku serwer bazy danych nie uwierzytelnia ID autoryzowanego użytkownika. Jednak korzystając z tej opcji, trzeba mieć pewność, że aplikacja nie pozwala na przełączanie na ID autoryzowanego użytkownika bez sprawdzenia poprawności i uwierzytelnienia tego ID autoryzowanego użytkownika. W przeciwnym razie w systemie powstanie luka w zabezpieczeniach.

Jawne zaufane połączenia mogą być tworzone i użytkownik może być w nich przełączany w przypadku połączeń za pośrednictwem produktu DB2 Connect z użyciem interfejsów CLI lub JDBC, łącznie z połączeniami nawiązanymi za pośrednictwem interfejsu XA. Tworzenie jawnego zaufanego połączenia i przełączanie użytkowników wymaga ustawienia specjalnych atrybutów połączenia. Oznacza to, że istniejące aplikacje będą musiały zostać zmodyfikowane, aby mogły korzystać z jawnych zaufanych połączeń.

Poza wymienionymi różnicami z zaufanego połączenia (niejawnego lub jawnego) można korzystać w ten sam sposób, jak ze zwykłego połączenia. Trzeba jednak pamiętać, aby w jawny sposób zakończyć jawne zaufane połączenie po zakończeniu pracy, nawet jeśli jest ono w stanie zerwania lub rozłączenia. W przeciwnym razie używane przez połączenie zasoby mogłyby nie zostać zwolnione. Ten problem nie istnieje w przypadku niejawnych zaufanych połączeń.

Uwaga:

1.

   Ważne: Przełączanie użytkowników bez podawania hasła powoduje ominięcie procedury uwierzytelniania na serwerze bazy danych. Aplikacja użytkownika nie może dopuścić do przełączenia na ID autoryzowanego użytkownika bez podania hasła, chyba że ten ID autoryzowanego użytkownika został już wcześniej sprawdzony i uwierzytelniony przez aplikację. W przeciwnym przypadku powstaje luka w zabezpieczeniach.

2. Jawne zaufane połączenia nie powinny korzystać z uwierzytelniania typu CLIENT. To zastrzeżenie nie dotyczy niejawnych zaufanych połączeń.
3. Aplikacje korzystające z jawnych zaufanych połączeń powinny być uruchamiane tylko na komputerach bezpiecznych, do których dostęp jest chroniony hasłem i możliwy tylko dla uprawnionych osób. To zastrzeżenie nie dotyczy niejawnych zaufanych połączeń.

Tworzenie i kończenie zaufanego połączenia przy pomocy interfejsu CLI

Jeśli serwer bazy danych, z którym nawiązywane jest połączenie, został odpowiednio skonfigurowany, korzystając z interfejsu CLI można utworzyć jawne zaufane połączenie.

W tej procedurze przyjęto, że menedżer transakcji XA nie jest używany. Jeśli menedżer transakcji XA jest używany, trzeba jedynie upewnić się, że został on skonfigurowany w taki sposób, aby przy wywoływaniu komendy xa_open ustawiał parametr konfiguracyjny TCTX na wartość TRUE. Jeśli tak jest, to dowolne połączenie, które może być jawnym zaufanym połączeniem, będzie tego typu połączeniem. Aby sprawdzić, czy połączenie jest jawnym zaufanym połączeniem, patrz krok 3.

- Baza danych, z którą ma nastąpić połączenie, musi obsługiwać zaufane konteksty.
- Zaufany kontekst musi być zdefiniowany tak, aby mógł rozpoznać klienta jako zaufanego.
- Konieczna jest znajomość ID autoryzowanego użytkownika, który został określony w zaufanym kontekście. Systemowy ID autoryzowanego użytkownika w zaufanym połączeniu to ID autoryzowanego użytkownika udostępniony serwerowi jako nazwa użytkownika podczas tworzenia połączenia. Aby połączenie użytkownika mogło być zaufane w określonym zaufanym kontekście, systemowy ID autoryzowanego użytkownika

musi być tym, który został podany w tym zaufanym kontekście. Poprawny systemowy ID autoryzowanego użytkownika i hasło do niego można uzyskać od administratora bezpieczeństwa.

W poniższych przykładowych instrukcjach użyto języka C i założono, że zmienna conn jest wskaźnikiem do poprawnego, lecz niepołączonego, uchwytu połączenia. Zmienna rc ma typ danych SQLRETURN.

1. Niezależnie od atrybutów połączenia konfigurowanych dla normalnego połączenia, ustaw atrybut połączenia SQL_ATTR_USE_TRUSTED_CONTEXT na wartość SQL_TRUE, wywołując funkcję SQLSetConnectAttr.

```
rc = SQLSetConnectAttr(
    conn,
    SQL_ATTR_USE_TRUSTED_CONTEXT, SQL_TRUE, SQL_IS_INTEGER
);
```

2. Połącz się z bazą danych w taki sam sposób, jak w przypadku zwykłego połączenia, na przykład wywołując funkcję SQLConnect. Jako nazwę użytkownika i hasło zastosuj systemowy ID autoryzowanego użytkownika i jego hasło. Sprawdź wystąpienie ewentualnych błędów i ostrzeżeń, szczególnie tych z tabeli Tabela 5.

Tabela 5. Błędy wskazujące niepowodzenie w utworzeniu zaufanego połączenia

SQLCODE	SQLSTATE	Znaczenie
SQL20360W	01679	Połączenie nie mogło być zrealizowane jako zaufane połączenie. W zamian zostało nawiązane zwykłe połączenie.

   Jeśli komunikaty o błędach i ostrzeżenia nie wskazują inaczej, połączenie jest nawiązane jako jawne zaufane połączenie.
3. (Opcjonalnie) Można sprawdzić, czy nawiązane połączenie jest jawnym zaufanym połączeniem przez sprawdzenie wartości atrybutu połączenia SQL_ATTR_USE_TRUSTED_CONTEXT przy pomocy funkcji SQLGetConnectAttr. Jeśli ma on wartość SQL_TRUE, połączenie jest jawnym zaufanym połączeniem.
4. Po zakończeniu pracy trzeba pamiętać, aby w jawny sposób zakończyć połączenie, nawet jeśli jest ono w stanie zerwania lub rozłączenia. Jeśli jawne zaufane połączenie nie zostanie w sposób jawny rozłączone, używane przez nie zasoby mogą nie zostać zwolnione.

Uwaga:

1. Jawne zaufane połączenia nie powinny korzystać z uwierzytelniania typu CLIENT. To zastrzeżenie nie dotyczy niejawnych zaufanych połączeń.
2. Aplikacje korzystające z jawnych zaufanych połączeń powinny być uruchamiane tylko na komputerach bezpiecznych, do których dostęp jest chroniony hasłem i możliwy tylko dla uprawnionych osób. To zastrzeżenie nie dotyczy niejawnych zaufanych połączeń.

Przełączanie użytkowników w zaufanym połączeniu poprzez interfejs CLI

W jawnym zaufanym połączeniu można przełączać użytkowników przez interfejs wiersza komend (interfejs CLI). Informacje na temat tego, czym jest przełączanie użytkowników przy użyciu zaufanego połączenia, można znaleźć w odpowiednim temacie w sekcji odsyłaczy do stron pokrewnych.

- Połączenie musi zostać pomyślnie utworzone jako jawne zaufane połączenie.
- Jawne zaufane połączenie nie może znajdować się wewnątrz transakcji.

- Zaufany kontekst, który umożliwił utworzenie jawnego zaufanego połączenia, musi być skonfigurowany w sposób pozwalający na przełączenie na wymagany ID autoryzowanego użytkownika.

W poniższych przykładowych instrukcjach użyto języka C i założono, że zmienna **conn** jest wskaźnikiem do nawiązanego jawnego zaufanego połączenia. Zmienna *rc* ma typ danych SQLRETURN. Przyjmuje się, że zmienna *newuser* jest wskaźnikiem do łańcucha znaków przechowującego ID autoryzowanego użytkownika, na który ma nastąpić przełączenie. Zmienna *passwd* jest wskaźnikiem do łańcucha znaków zawierającego hasło dla tego ID autoryzowanego użytkownika.

1. Wywołaj funkcję SQLSetConnectAttr, aby ustawić atrybut SQL_ATTR_TRUSTED_CONTEXT_USERID. Wpisz do niego ID autoryzowanego użytkownika, na który ma nastąpić przełączenie.

   ```
   rc = SQLSetConnectAttr(
       conn,
       SQL_ATTR_TRUSTED_CONTEXT_USERID, newuser, SQL_NTS
   );
   //Sprawdź błędy
   ```

   Sprawdź wystąpienie ewentualnych błędów i ostrzeżeń, szczególnie tych z tabeli Tabela 6.

Tabela 6. Błędy wskazujące niepowodzenie ustawienia nowego ID autoryzowanego użytkownika podczas przełączania użytkowników

SQLCODE	Znaczenie
CLI0106E	Połączenie nie jest nawiązane.
CLI0197E	Połączenie nie jest zaufanym połączeniem.
CLI0124E	Wystąpił problem związany z podaną wartością. Sprawdź, czy nie jest to wartość NULL, czy nie jest ona zbyt długa itp.
CLI0196E	Połączenie bierze udział w jednostce pracy, która nie pozwala na przełączanie użytkowników. Aby przełączać użytkowników, połączenie nie może wykonywać transakcji.

2. (Opcjonalne, chyba że zaufany kontekst, który utworzył to zaufane połączenie, wymaga podania hasła dla ID autoryzowanego użytkownika, na który następuje przełączenie.) Wywołaj funkcję SQLSetConnectAttr, aby ustawić atrybut SQL_ATTR_TRUSTED_CONTEXT_PASSWORD. Wpisz do niego hasło dla nowego ID autoryzowanego użytkownika.

   ```
   rc = SQLSetConnectAttr(
       conn,
       SQL_ATTR_TRUSTED_CONTEXT_PASSWORD, passwd, SQL_NTS
   );
   //Sprawdź błędy
   ```

   Sprawdź wystąpienie ewentualnych błędów i ostrzeżeń, zarówno tych z tabeli Tabela 6, jak i tych z tabeli Tabela 7.

Tabela 7. Błędy wskazujące niepowodzenie ustawienia hasła podczas przełączania użytkowników

SQLCODE	Znaczenie
CLI0198E	Atrybut SQL_ATTR_TRUSTED_CONTEXT_USERID nie został jeszcze ustawiony.

3. Postępuj tak, jak przy zwykłym połączeniu. Jeśli korzystasz z menedżera transakcji XA, przełączenie użytkownika jest wykonywane jako część następnego żądania, w przeciwnym razie przełączenie użytkownika jest wykonywane tuż przed zainicjowaniem

kolejnego wywołania funkcji, która uzyskuje dostęp do bazy danych (na przykład SQLExecDirect). W obu tych przypadkach poza błędami i ostrzeżeniami, których wystąpienie należy zazwyczaj sprawdzić, sprawdź dodatkowo także błędy wymienione w tabeli Tabela 8. Błędy zawarte w tabeli Tabela 8 wskazują, że przełączenie użytkownika nie powiodło się.

Tabela 8. Błędy wskazujące na niepowodzenie przełączania użytkowników

SQLCODE	**Znaczenie**
SQL1046N	Zaufany kontekst, który umożliwił nawiązanie zaufanego połączenia, nie jest skonfigurowany w sposób pozwalający na przełączenie na wymagany ID autoryzowanego użytkownika. Nie można przełączyć na ten ID autoryzowanego użytkownika, dopóki nie zostanie zmieniony zaufany kontekst.
SQL30082N	Podane hasło nie jest poprawne dla ID autoryzowanego użytkownika, na który następuje przełączenie.
SQL0969N z błędem rodzimym -20361	Istnieje pewne ograniczenie na poziomie bazy danych, które uniemożliwia przełączenie na tego użytkownika.

Jeśli przełączenie użytkownika nie powiedzie się, połączenie pozostanie w stanie niepodłączonym do czasu pomyślnego przełączenia. W stanie niepodłączonym zaufanego połączenia można przełączać użytkowników, ale nie można uzyskać dostępu do serwera bazy danych. Połączenie pozostanie w stanie niepodłączonym, dopóki nie nastąpi pomyślne przełączenie użytkowników.

Uwagi:

1. **Ważne:** Przełączanie użytkowników bez podawania hasła powoduje ominięcie procedury uwierzytelniania na serwerze bazy danych. Aplikacja użytkownika nie może dopuścić do przełączenia na ID autoryzowanego użytkownika bez podania hasła, chyba że ten ID autoryzowanego użytkownika został już wcześniej sprawdzony i uwierzytelniony przez aplikację. W przeciwnym przypadku powstaje luka w zabezpieczeniach.
2. Podanie wartości NULL dla atrybutu SQL_ATTR_TRUSTED_CONTEXT_USERID jest równoważne określeniu systemowego ID autoryzowanego użytkownika zaufanego kontekstu (identyfikator użytkownika użyty przy tworzeniu jawnego zaufanego połączenia).
3. Po pomyślnym ustawieniu wartości atrybutu połączenia SQL_ATTR_TRUSTED_CONTEXT_USERID w jawnym zaufanym połączeniu połączenie to jest natychmiast resetowane. Skutkiem resetowania jest sytuacja identyczna, jak po utworzeniu nowego połączenia przy użyciu początkowych atrybutów tego połączenia. Ta operacja resetowania występuje nawet wtedy, gdy wartość nadana atrybutowi połączenia to systemowy ID autoryzowanego użytkownika, wartość NULL lub taka sama wartość, jak bieżąca wartość atrybutu.
4. Jeśli ustawiony jest atrybut SQL_ATTR_TRUSTED_CONTEXT_PASSWORD, hasło zostanie uwierzytelnione podczas przetwarzania przełączania użytkowników, nawet jeśli zaufany kontekst, który umożliwił utworzenie zaufanego połączenia, nie wymaga uwierzytelniania podczas przełączania użytkownika na dany ID autoryzowanego użytkownika. Skutkiem tego jest niepotrzebny narzut. Ta reguła nie ma zastosowania do systemowego ID autoryzowanego użytkownika zaufanego kontekstu. Jeśli systemowy ID autoryzowanego użytkownika zaufanego kontekstu nie wymaga uwierzytelnienia podczas przełączania na niego, to nie zostanie uwierzytelniony nawet w sytuacji, gdy podano hasło.

Uwagi dotyczące uwierzytelniania w programie DB2 Connect

Administrator programu DB2 Connect może wspólnie z administratorem bazy danych na serwerze zSeries lub System i określić miejsce, w którym będzie sprawdzana poprawność nazw użytkowników i haseł:

- po stronie klienta
- na serwerze zSeries lub System i
- pojedyncze logowanie oraz sprawdzanie za pomocą systemów pochodzących od innych producentów (Kerberos)

Uwaga: Jeśli zdalny klient nie określił typu uwierzytelniania, zostanie zastosowany typ domyślny SERVER_ENCRYPT. Jeśli ten typ nie jest akceptowany przez serwer, klient będzie próbował ponowić połączenie, używając odpowiedniej wartości zwróconej przez serwer. Mając na uwadze optymalizację wydajności, należy zawsze określać typ uwierzytelniania na kliencie, co pozwoli wyeliminować dodatkowy ruch w sieci.

Począwszy od wersji 8.2.2 produktu DB2 Connect (odpowiednik wersji 8.1 z pakietem poprawek 9) gateway nie jest już biernym uczestnikiem negocjacji uwierzytelniania. Zamiast tego odgrywa aktywną rolę. Określony we wpisie katalogu bazy danych gatewaya typ uwierzytelniania przesłania typ uwierzytelniania zapisany w katalogu po stronie klienta. Po stronie klienta, gatewaya i serwera muszą być określone zgodne typy. Jeśli zapisany w katalogu gatewaya typ uwierzytelniania nie został podany we wpisie katalogu bazy danych, domyślnym typem uwierzytelniania wymaganym od serwera będzie typ SERVER. Jeśli jednak serwer nie obsługuje typu uwierzytelniania SERVER, między klientem i serwerem nadal zachodzić będzie negocjacja. Takie zachowanie różni się od zachowania po stronie klienta, który przyjmuje wartość domyślną SERVER_ENCRYPT, jeśli typ uwierzytelniania nie został określony.

Zapisany w katalogu gatewaya typ uwierzytelniania nie jest używany, jeśli opcja DB2NODE lub SQL_CONNECT_NODE interfejsu Set Client API została ustawiona po stronie klienta. W takich przypadkach negocjacje odbywają się wyłącznie pomiędzy klientem a serwerem.

Z produktem DB2 Connect dozwolone są następujące typy uwierzytelniania:

CLIENT
Poprawność nazwy użytkownika i hasła jest sprawdzana po stronie klienta.

SERVER
Nazwa i hasło użytkownika są sprawdzane przez serwer bazy danych na serwerze zSeries lub System i.

SERVER_ENCRYPT
Tak jak w przypadku uwierzytelniania typu SERVER, nazwa i hasło użytkownika są sprawdzane przez serwer bazy danych na serwerze zSeries lub System i, ale przed przesłaniem identyfikatory i hasła są szyfrowane po stronie klienta.

DATA_ENCRYPT
Umożliwia szyfrowanie danych użytkownika podczas komunikacji między klientem i serwerem. Uwierzytelnianie tego typu nie jest obsługiwane na serwerze bazy danych System i.

KERBEROS
Umożliwia klientowi logowanie się na serwerze za pomocą uwierzytelniania Kerberos, zamiast za pomocą tradycyjnej kombinacji identyfikatora i hasła. Ten typ uwierzytelniania wymaga, aby protokół Kerberos był obsługiwany zarówno przez klienta, jak i przez serwer.

SERVER_ENCRYPT_AES
Nazwa i hasło użytkownika są sprawdzane przez serwer bazy danych na serwerze zSeries, ale przed przesłaniem identyfikatory i hasła są szyfrowane po stronie klienta przy użyciu algorytmu AES. Ten typ uwierzytelniania jest dostępny począwszy od pakietu poprawek 3 dla programu DB2 Connect wersja 9.5.

Uwierzytelnianie typu Kerberos jest unikalne dzięki temu, że klient nie przekazuje identyfikatora ani hasła użytkownika bezpośrednio do serwera. Zamiast tego protokół Kerberos działa jako niezależny mechanizm uwierzytelniania. Użytkownik wprowadza identyfikator i hasło tylko raz, na terminalu klienta, a protokół Kerberos sprawdza poprawność tego podpisu. Po wykonaniu tej czynności protokół Kerberos w sposób automatyczny i bezpieczny przekazuje autoryzację użytkownika dowolnym żądanym usługom sieciowym lub lokalnym. Oznacza to, że użytkownik nie musi ponownie wprowadzać identyfikatora i hasła, aby zalogować się do zdalnego serwera DB2. Możliwość pojedynczego logowania udostępniana poprzez uwierzytelnianie Kerberos wymaga, aby zarówno produkt DB2 Connect, jak i serwer bazy danych, z którym nawiązywane jest połączenie, udostępniał obsługę protokołu Kerberos.

Uwaga: Typ uwierzytelniania GSSPLUGIN nie jest obsługiwany.

Obsługa protokołu Kerberos

Warstwa uwierzytelniania Kerberos, która obsługuje system biletowania, jest zintegrowana w ramach mechanizmu Active Directory systemu Windows 2000. Strony klienta i serwera aplikacji komunikują się odpowiednio z modułami klienta i serwera Kerberos SSP (Security Support Provider, dostawca usług zabezpieczeń). Interfejs Security Support Provider Interface (SSPI) stanowi interfejs wysokiego poziomu protokołu Kerberos SSP i innych protokołów bezpieczeństwa.

Typowa konfiguracja

Aby skonfigurować program DB2 z uwierzytelnianiem Kerberos, należy przygotować:

- strategię uwierzytelniania dla programu DB2 (jako usługi) w katalogu Active Directory współużytkowanym w sieci,
- związek zaufania między centrami dystrybucji kluczy Kerberos Key Distribution Center (KDC).

W najprostszym scenariuszu istnieje przynajmniej jedna zaufana relacja KDC do skonfigurowania, którą jest relacja pomiędzy sterowaną przez KDC stacją roboczą klienta a systemem System i, OS/390 lub z/OS. System operacyjny OS/390 w wersji 2 wydanie 10 lub z/OS w wersji 1 wydanie 2 udostępnia przetwarzanie biletów Kerberos za pośrednictwem narzędzia RACF, które umożliwia hostowi występowanie w roli KDC systemu UNIX.

Produkt DB2 Connect udostępnia jak zwykle funkcjonalność routera w ustawieniu trójwarstwowym. Program ten nie bierze udziału w uwierzytelnianiu, gdy zabezpieczenia realizowana są za pomocą protokołu Kerberos. Przekazuje tylko znacznik bezpieczeństwa klienta do produktu DB2 for i5/OS lub DB2 for OS/390 and z/OS. Nie ma potrzeby, aby gateway produktu DB2 Connect był elementem dziedziny Kerberos klienta lub hosta.

Kompatybilność wsteczna

Minimalne wymagania produktu DB2 do obsługi protokołu Kerberos:

IBM Data Server Client:
Wersja 8

DB2 Connect:
Wersja 8

DB2 Universal Database (UDB) for OS/390 and z/OS:
Wersja 7

Wskazówki dotyczące zabezpieczeń w systemach OS/390 i z/OS

W tym temacie przedstawiono wskazówki dotyczące bezpieczeństwa połączenia produktu DB2 Connect z serwerem bazy danych DB2 for OS/390 and z/OS.

Pole rozszerzonych zabezpieczeń

Upewnij się, że pole rozszerzonych zabezpieczeń produktu DB2 for OS/390 and z/OS ma wartość YES. To pole znajduje się na panelu DSNTIPR produktu DB2 for OS/390 and z/OS.

Kody rozszerzonych zabezpieczeń

Do czasu powstania produktu DB2 Universal Database for z/OS and OS/390 wersja 5.1 żądania połączenia dostarczające identyfikator użytkownika lub hasło mogły zakończyć się niepowodzeniem z kodem przyczyny SQL30082 równym 0, ale bez żadnej innej wskazówki określającej przyczynę wystąpienia błędu.

W programie DB2 Universal Database for z/OS and OS/390 wersja 5.1 wprowadzono rozszerzenie umożliwiające obsługę kodów rozszerzonych zabezpieczeń. Określenie rozszerzonych zabezpieczeń stwarza dodatkowe możliwości diagnostyki, takie jak opcja (PASSWORD EXPIRED), dodane do kodu przyczyny.

Aby korzystać z tych opcji, dla parametru instalacyjnego DB2 Universal Database for z/OS and OS/390 ZPARM dla rozszerzonych zabezpieczeń należy ustawić wartość YES. Należy użyć ekranu instalacji DB2 Universal Database for z/OS and OS/390 DSN6SYSP, aby ustawić EXTSEC=YES. Można do tego celu użyć również ekranu 1 DDF (DSNTIPR). Wartością domyślną jest EXTSEC=NO. W przypadku utraty ważności hasła system Windows, Linux, UNIX oraz aplikacje WWW korzystające z produktu DB2 Connect otrzymają komunikat o błędzie SQL30082.

sprawdzone wcześniej zabezpieczenia TCP/IP

Aby zapewnić obsługę dla opcji bezpieczeństwa AUTHENTICATION=CLIENT produktu DB2, skorzystaj z panelu instalacyjnego DSNTIP4 (DDF panel 2) produktu DB2 Universal Database for z/OS and OS/390 w celu ustawienia zweryfikowanych już zabezpieczeń protokołu TCP/IP na wartość YES.

Zabezpieczenia aplikacji ODBC i Java

Aplikacje ODBC i Java stacji roboczej używają dynamicznego języka SQL. W niektórych instalacjach może to być źródłem pewnych problemów związanych z bezpieczeństwem. Program DB2 Universal Database for z/OS and OS/390 wprowadza nową opcję wiązania DYNAMICRULES(BIND) umożliwiającą wykonanie dynamicznego SQL z autoryzacją właściciela lub konsolidatora.

Programy DB2 i DB2 Connect wersja 5 dostarczają nowego parametru konfiguracyjnego CLI/ODBC CURRENTPACKAGESET w pliku konfiguracyjnym DB2CLI.INI. Należy mu

nadać nazwę schematu, który ma odpowiednie uprawnienia. Instrukcja SQL SET CURRENT PACKAGESET schemat będzie automatycznie wywoływana dla aplikacji po każdym połączeniu.

Aby zaktualizować DB2CLI.INI, należy użyć programu ODBC Manager.

Obsługa zmiany hasła

Jeśli hasło użytkownika utraciło ważność, instrukcja CONNECT języka SQL zwraca komunikat o błędzie, taki jak SQLCODE -30082 z kodem przyczyny 1. Korzystając z produktu DB2 Connect, można zdalnie zmienić hasło. Wykorzystując architekturę DRDA, produkt DB2 Universal Database for z/OS and OS/390 może zmienić hasło, wydając następującą instrukcję CONNECT:

```
CONNECT TO <baza danych> USER <ID użytkownika> USING <hasło>
   NEW <nowe_hasło> CONFIRM <nowe_hasło>
```

Do zmiany hasła można również użyć okna dialogowego Asysty podczas konfigurowania DB2 o nazwie Zmiana hasła.

Typy uwierzytelniania obsługiwane w programie DB2 Connect

Typy uwierzytelniania obsługiwane w programie DB2 Connect to CLIENT, SERVER, SERVER_ENCRYPT, KERBEROS, DATA_ENCRYPT oraz SERVER_ENCRYPT_AES.

Typy zabezpieczeń dla połączeń TCP/IP

Protokół TCP/IP nie zawiera opcji zabezpieczeń w warstwie protokołu sieciowego. Typ uwierzytelnienia określa miejsce przeprowadzania uwierzytelnienia. W produkcie DB2 Connect obsługiwane są tylko kombinacje przedstawione w poniższej tabeli. Ustawienie typu uwierzytelniania zapisane jest we wpisie katalogu bazy danych na serwerze DB2 Connect.

Tabela 9. Poprawne scenariusze bezpieczeństwa

Scenariusz	Typ uwierzytelniania	Sprawdzanie
1	CLIENT	Klient
2	SERVER	Serwer bazy danych w systemie zSeries lub System i
3	SERVER_ENCRYPT	Serwer bazy danych w systemie zSeries lub System i
4	KERBEROS	Zabezpieczenia Kerberos
5	DATA_ENCRYPT	zSeries
6	SERVER_ENCRYPT_AES	Serwer bazy danych w systemie zSeries

Omówienie typów zabezpieczeń

Poniższe omówienie odnosi się do połączeń opisanych powyżej i wymienionych w tabeli Tabela 9. Każdy scenariusz został poniżej opisany bardziej szczegółowo:

- W scenariuszu 1 nazwa i hasło użytkownika są sprawdzane tylko na kliencie zdalnym. W wypadku klienta lokalnego nazwa użytkownika i hasło są sprawdzane tylko na serwerze DB2 Connect.

  Użytkownik jest uwierzytelniany w systemie, w którym się zalogował. Przez sieć jest wysyłany identyfikator użytkownika, ale nie hasło. Z tego rodzaju zabezpieczeń można korzystać, jeśli wszystkie klienckie stacje robocze mają odpowiednie metody zabezpieczeń, którym można zaufać.
- W scenariuszu 2 nazwa i hasło użytkownika są sprawdzane tylko przez serwer bazy danych na serwerze zSeries lub System i. Identyfikator i hasło użytkownika

są przesyłane przez sieć ze zdalnego klienta do serwera DB2 Connect, a następnie z serwera DB2 Connect do serwera bazy danych w systemie zSeries lub System i.

- Scenariusz 3 jest taki sam, jak scenariusz 2, za wyjątkiem faktu, że ID użytkownika i hasło są zaszyfrowane.
- W scenariuszu 4 klient uzyskuje bilet Kerberos z KDC Kerberos. Bilet jest przekazywany w niezmienionej postaci przez program DB2 Connect do serwera, który go sprawdza.
- Scenariusz 5 jest taki sam, jak scenariusz 3, za wyjątkiem faktu, że dane użytkownika również są również zaszyfrowane, a DATA_ENCRYPT nie obsługuje serwera bazy danych System i.
- Scenariusz 6 jest taki sam, jak scenariusz 3, ale jest w nim wykorzystywany algorytm szyfrowania AES (Advanced Encryption Standard). Ten typ uwierzytelniania jest obsługiwany począwszy od pakietu poprawek 3 dla programu DB2 Connect wersja 9.5.

Rozdział 6. Wiązanie aplikacji i programów narzędziowych (DB2 Connect)

Aplikacje zawierające wbudowany SQL muszą być powiązane ze wszystkimi bazami danych, z których korzystają. Na platformach, na których dostępne są odpowiednie funkcje, można to zrobić, korzystając z Centrum komend i Asysty podczas konfigurowania.

Wystarczy raz powiązać aplikację z odpowiednimi bazami danych. Podczas wykonywania wiązania dla każdej instrukcji SQL zapamiętywane są plany dostępu do bazy danych. Dostarczane są one przez programistów aplikacji. Plany te zawarte są w *plikach powiązań* tworzonych podczas prekompilacji. Wiązanie to proces przetwarzania tych plików powiązań przez serwer bazy danych na hoście lub serwerze System i.

Ponieważ kilka programów narzędziowych dostarczanych wraz z oprogramowaniem DB2 Connect zostało napisanych przy użyciu wbudowanego SQL, muszą one zostać powiązane z serwerem bazy danych na hoście lub serwerze System i, aby można było z nich korzystać w danym systemie. Jeśli programy narzędziowe i interfejsy DB2 Connect nie są używane, nie ma konieczności wiązania ich z każdym serwerem bazy danych na hoście lub serwerze System i. Listy plików powiązań, wymaganych przez te programy narzędziowe, zawierają następujące pliki:

- ddcsmvs.lst dla platformy OS/390 lub z/OS
- ddcsvse.lst dla VSE
- ddcsvm.lst dla VM
- ddcs400.lst dla platformy OS/400

Powiązanie jednej z powyższych list plików z bazą danych jest równoznaczne z powiązaniem wszystkich narzędzi z tą bazą danych.

Jeśli produkt serwerowy DB2 Connect jest zainstalowany, narzędzia DB2 Connect muszą zostać powiązane z każdym hostem lub serwerem bazy danych System i przed ich użyciem z tym systemem. Jeśli klienci mają zainstalowaną tę samą wersję pakietu poprawek, wystarczy powiązać narzędzia tylko raz, niezależnie od liczby platform klientów.

Na przykład jeśli jest 10 klientów systemu Windows i 10 klientów systemu AIX łączących się z produktem DB2 Universal Database (UDB) for OS/390 and z/OS przez produkt DB2 Connect Enterprise Edition na serwerze Windows, należy wykonać jedną z poniższych czynności:

- Powiązać listę ddcsmvs.lst na jednym z klientów systemu Windows.
- Powiązać listę ddcsmvs.lst na jednym z klientów systemu AIX.
- Powiązać listę ddcsmvs.lst na serwerze DB2 Connect.

Przyjęto następujące założenia:

- Wszystkie klienty mają ten sam poziom usług. W przeciwnym razie należałoby powiązać każdego klienta na odpowiednim poziomie usług.
- Serwer ma ten sam poziom usług, co klienty. W przeciwnym razie należy powiązać także serwer.

Oprócz narzędzi DB2 Connect wszystkie inne aplikacje korzystające z wbudowanego SQL muszą być powiązane z odpowiednimi bazami danych. Aplikacja niepowiązana podczas uruchamiania generuje zazwyczaj komunikat o błędzie SQL0805N. Można utworzyć własne pliki list powiązań dla wszystkich aplikacji, które muszą być powiązane.

© Copyright IBM Corp. 1993, 2010

Dla każdego serwera bazy danych na hoście lub serwerze System i, z którym następuje wiązanie, należy wykonać następujące czynności:

1. Uzyskaj wymagane uprawnienia do systemu zarządzania serwerem bazy danych na hoście lub serwerze System i:

   OS/390 lub z/OS

   Wymagane autoryzacje to:

   - SYSADM lub
   - SYSCTRL lub
   - BINDADD *oraz* CREATE IN COLLECTION NULLID.

   Uwaga: Uprawnienia BINDADD i CREATE IN COLLECTION NULLID są wystarczające **tylko** wtedy, gdy pakiety jeszcze nie istnieją. Na przykład wtedy, gdy tworzy się je po raz pierwszy.

   Jeśli pakiety już istnieją i wykonuje się ponowne wiązanie, wymagane uprawnienia zależą od tego, kto wykonał pierwsze wiązanie.

   A) Jeśli oba wiązania wykonała ta sama osoba, wystarczy którekolwiek z powyższych uprawnień.

   B) Jeśli oryginalne wiązanie zostało wykonane przez inną osobę, to podczas wykonywania drugiego wiązania konieczne są uprawnienia SYSADM lub SYSCTRL. W tym przypadku posiadanie uprawnień BINDADD i CREATE IN COLLECTION NULLID nie umożliwia wykonania wiązania. Aby utworzyć pakiet, nie są jednak konieczne uprawnienia SYSADM czy SYSCTRL. Należy mieć uprawnienie BIND dla każdego z istniejących pakietów, który ma zostać zastąpiony.

   VSE lub VM

   Wymagana autoryzacja to DBA. Jeśli używana jest opcja GRANT w komendzie wiązania (w celu uniknięcia konieczności przyznawania dostępu do każdego pakietu DB2 Connect osobno), identyfikator użytkownika NULLID musi mieć uprawnienia do przyznawania uprawnień innym użytkownikom, odnoszące się do następujących tabel:

   - system.syscatalog
   - system.syscolumns
   - system.sysindexes
   - system.systabauth
   - system.syskeycols
   - system.syssynonyms
   - system.syskeys
   - system.syscolauth
   - system.sysuserauth

   W systemie VSE lub VM można uruchomić komendę:

   ```
   grant select on tabela to nullid with grant option
   ```

   OS/400

   Uprawnienie *CHANGE lub wyższe w kolekcji NULLID.

2. Uruchom komendy podobne do następujących:

   ```
   db2 connect to DBALIAS user USERID using PASSWORD
   db2 bind ścieżka@ddcsmvs.lst blocking all
         sqlerror continue messages ddcsmvs.msg grant public
   db2 connect reset
   ```

Gdzie parametry *DBALIAS*, *USERID* i *PASSWORD* dotyczą serwera bazy danych na hoście lub serwerze System i, ddcsmvs.lst jest plikiem listy powiązań dla systemu z/OS, a *ścieżka* określa położenie pliku listy powiązań.

Na przykład ścieżka *napęd*:\sqllib\bnd\ dotyczy wszystkich systemów operacyjnych Windows, a katalog *INSTHOME*/sqllib/bnd/ dotyczy wszystkich systemów operacyjnych Linux i UNIX, gdzie *napęd* określa dysk logiczny, na którym zainstalowano program DB2 Connect, natomiast *INSTHOME* określa katalog osobisty instancji DB2 Connect.

W komendzie bind można użyć opcji nadawania, aby nadać uprawnienie EXECUTE wszystkim użytkownikom (PUBLIC), użytkownikowi określonemu przez nazwę lub grupie określonej identyfikatorem. Jeśli w komendzie bind nie zostanie użyta opcja nadawania, należy oddzielnie uruchomić komendę GRANT EXECUTE (RUN).

Aby poznać nazwy pakietów dla plików powiązań, należy wprowadzić następującą komendę:

```
ddcspkgn @bindfile.lst
```

Na przykład:

```
ddcspkgn @ddcsmvs.lst
```

można otrzymać następujący wynik:

```
Plik powiązań                  Nazwa pakietu
------------------------------ ------------------------------
f:\sqllib\bnd\db2ajgrt.bnd     SQLAB6D3
```

Aby określić odpowiednie wartości dla programu DB2 Connect, należy uruchomić narzędzie *ddcspkgn*, na przykład:

```
ddcspkgn @ddcsmvs.lst
```

Tego programu narzędziowego można także użyć do określenia nazwy pakietu dla poszczególnych plików powiązań, na przykład:

```
ddcspkgn bindfile.bnd
```

Uwaga:

a. Wymagane jest użycie opcji wiązania sqlerror continue; ta opcja jest automatycznie wybrana w przypadku wiązania aplikacji z użyciem narzędzi DB2 lub procesora wiersza komend (CLP). Zamienia ona błędy wiązania w ostrzeżenia, dzięki czemu można utworzyć pakiet również dla pliku zawierającego błędy. Umożliwia to z kolei użycie jednego pliku powiązania do wielu serwerów, nawet wtedy, gdy jedna implementacja serwera uzna składnię SQL innej implementacji za niepoprawną. Dlatego wiążąc dowolny z plików list ddcsxxx.lst z dowolnym serwerem bazy danych na hoście lub serwerze System i, należy się spodziewać generowania ostrzeżeń.

b. Łącząc się z bazą danych DB2 przez program DB2 Connect, należy korzystać z listy powiązań db2ubind.lst i nie używać parametru sqlerror continue, który jest poprawny tylko dla połączeń z serwerem bazy danych na hoście lub serwerze System i. Ponadto do połączeń z bazą danych DB2 zalecane jest korzystanie z klientów DB2 dostarczanych z programem DB2, a nie z DB2 Connect.

3. Podobnych instrukcji należy użyć, aby powiązać każdą aplikację lub listę aplikacji.
4. Jeśli w systemie występują zdalne klienty z poprzedniego wydania programu DB2, może zajść konieczność powiązania narzędzi tych klientów z programem DB2 Connect.

Rozdział 7. Aktualizacje na wielu serwerach

Aktualizacja na wielu serwerach, znana również jako rozproszona jednostka pracy (DUOW, distributed unit of work) lub zatwierdzanie dwufazowe, to funkcja umożliwiająca aplikacji aktualizację danych na wielu zdalnych serwerach baz danych z zachowaniem integralności danych. Na przykład może być to transakcja bankowa polegająca na transferze pieniędzy z jednego konta na inne, znajdujące się na innym serwerze baz danych.

W przypadku takiej transakcji najważniejszą kwestią jest to, aby aktualizacje, które polegają na zaimplementowaniu operacji debetowania jednego konta, nie zostały zatwierdzone, jeśli jednocześnie nie zostaną zatwierdzone aktualizacje operacji kredytowania drugiego konta. Aktualizacja na wielu serwerach odnosi się do sytuacji, gdy dane reprezentujące konta są zarządzane przez dwa różne serwery baz danych.

Produkty DB2 zapewniają wszechstronną obsługę aktualizacji na wielu serwerach. Obsługa ta jest dostępna dla aplikacji utworzonych przy użyciu zwykłego języka SQL, a także dla aplikacji korzystających z monitorów przetwarzania transakcyjnego (monitorów TP), w których zaimplementowano specyfikację interfejsu XA X/Open. Przykładami takich monitorów TP są produkty IBM TxSeries CICS, IBM Message and Queuing Series, IBM Component Broker Series, IBM San Francisco Project oraz Microsoft Transaction Server (MTS), BEA Tuxedo i inne. W zależności od tego, czy wykorzystywane są aktualizacje na wielu serwerach w rodzimym języku SQL, czy aktualizacje na wielu serwerach za pomocą monitora TP, istnieją różne wymagania dotyczące konfiguracji.

Obie metody mogą też używać instrukcji SQL Connect do wskazywania bazy danych, której dotyczyć mają kolejne instrukcje SQL. Jeśli żaden monitor TP nie poinformuje produktu DB2, że będzie koordynował transakcję (wskazane przez odebranie w produkcie DB2 wywołania xa_open z monitora TP w celu nawiązania połączenia z bazą danych), to do koordynowania tej transakcji zostanie użyte oprogramowanie DB2.

Wykonując aktualizację na wielu serwerach poprzez monitor TP, aplikacja musi żądać zatwierdzenia lub wycofania zmian, stosując interfejs API monitora TP, na przykład CICS SYNCPOINT, MTS SetAbort(). Jeśli aktualizacja na wielu serwerach przeprowadzana jest przy użyciu języka SQL, stosowane są zwykłe komendy SQL COMMIT i ROLLBACK.

Aktualizacja na wielu serwerach poprzez monitor TP może koordynować transakcję, która uzyskuje dostęp zarówno do produktu DB2, jak i menedżerów zasobów innych od DB2, takich jak Oracle, Informix lub SQLServer. Aktualizacja na wielu serwerach przy użyciu rodzimego języka SQL jest stosowana tylko w odniesieniu do serwerów DB2.

Aby transakcja aktualizacji na wielu serwerach mogła zadziałać, każda z baz danych uczestniczących w rozproszonej transakcji musi mieć możliwość obsługi rozproszonej jednostki pracy (jednostka DUOW). W tej chwili następujące serwery DB2 umożliwiają obsługę jednostek DUOW, dzięki czemu mogą brać udział w transakcjach rozproszonych:

- DB2 for Linux, UNIX and Windows w wersji 8 lub nowszej
- DB2 Universal Database (UDB) for OS/390 and z/OS, wersja 7
- DB2 for z/OS, wersja 8
- DB2 for i5/OS wymaga systemu operacyjnego OS/400 V5R1 lub nowszego

Transakcja rozproszona może zaktualizować dowolną kombinację serwerów baz danych, które są obsługiwane. Na przykład aplikacja użytkownika w ramach pojedynczej transakcji

© Copyright IBM Corp. 1993, 2010

może aktualizować kilka tabel w bazie danych DB2 w systemie operacyjnym Windows, w bazie danych DB2 for OS/390 and z/OS i w bazie danych DB2 for i5/OS.

Włączanie aktualizacji na wielu serwerach przy użyciu Centrum sterowania

Aby włączyć aktualizacje na wielu serwerach, można skorzystać z Centrum sterowania.

Aby włączyć aktualizację na wielu serwerach:

1. Uruchom Centrum sterowania.
2. Kliknij znak [+], aby rozwinąć widok drzewa.
3. Prawym przyciskiem myszy wskaż instancję, która ma zostać skonfigurowana. Zostanie otwarte menu podręczne.
4. Wybierz z menu opcje: **Aktualizacja na wielu serwerach —> Konfiguruj**. Otwarty zostanie Kreator aktualizacji na wielu serwerach.
5. Wybierz opcję **Użyj następującego monitora przetwarzania transakcyjnego** i określ monitor przetwarzania transakcyjnego. Pole to pokazuje wartości domyślne dla aktywnego monitora przetwarzania transakcyjnego. Jeśli nie chcesz używać monitora TP, wybierz opcję **Nie używaj monitora przetwarzania transakcyjnego**.

   Kliknij przycisk **Dalej**.
6. Jeśli używany jest monitor TP, określ ustawienia menedżera punktów synchronizacji. Jeśli monitor TP nie jest używany, określ bazę danych menedżera transakcji.
7. Kliknij przycisk **Zakończ**.

Testowanie aktualizacji na wielu serwerach przy użyciu Centrum sterowania

Konfigurację aktualizacji na wielu serwerach można przetestować przy użyciu Centrum sterowania.

Aby przetestować aktualizację na wielu serwerach:

1. Kliknij instancję prawym przyciskiem myszy, a następnie z menu podręcznego wybierz opcje: **Aktualizacja na wielu serwerach —> Testuj**. Zostanie otwarte okno Testuj aktualizację na wielu serwerach.
2. W oknie listy **Dostępne** spośród dostępnych baz danych wybierz bazy danych, które chcesz przetestować. Aby przenieść wybrane bazy z listy **Wybrane** lub na tę listę, możesz użyć przycisków strzałek (> i >>), które znajdują się na środku. Możesz także zmienić wybrany identyfikator użytkownika i hasło, edytując je bezpośrednio w oknie listy **Wybrane**.
3. Po dokonaniu wyboru kliknij przycisk **OK**. Zostanie otwarte okno z wynikami testu aktualizacji na wielu serwerach.
4. W oknie z wynikami testu aktualizacji na wielu serwerach znajdują się informacje o tym, dla których z wybranych baz danych test aktualizacji zakończył się pomyślnie, a dla których się nie powiódł. Okno zawiera kody SQL i komunikaty o błędach dla baz, dla których test się nie powiódł. Kliknij przycisk **Zamknij**, aby zamknąć okno.
5. Kliknij przycisk **Zamknij**, aby zamknąć okno Testuj aktualizację na wielu serwerach.

Aktualizacja na wielu serwerach i menedżer punktów synchronizacji

Host i serwery bazy danych System i wymagają udziału produktu DB2 Connect w rozproszonej transakcji pochodzącej z systemów Linux, Windows, UNIX lub aplikacji WWW. Ponadto wiele scenariuszy aktualizacji na wielu serwerach, w których uczestniczą serwery baz danych na hostach lub serwerach System i, wymaga skonfigurowania menedżera punktów synchronizacji (SPM, sync point manager). Po utworzeniu instancji DB2 menedżer DB2 SPM jest automatycznie konfigurowany z ustawieniami domyślnymi.

Konieczność użycia menedżera punktów synchronizacji wynika z wybranego protokołu (TCP/IP) i wykorzystania monitora przetwarzania transakcyjnego. Następująca tabela zawiera podsumowanie scenariuszy, które wymagają użycia menedżera punktów synchronizacji. Pokazuje ona również czy, produkt DB2 Connect jest wymagany przy dowolnym dostępie do hosta lub systemu System i z komputerów platformy Intel® lub UNIX. W wypadku aktualizacji na wielu serwerach komponent SPM programu DB2 Connect jest wymagany, jeśli używany jest monitor przetwarzania transakcyjnego.

Tabela 10. Scenariusze aktualizacji na wielu serwerach, w których wymagany jest składnik SPM – TCP/IP.

Czy używany jest monitor przetwarzania transakcyjnego (TP)?	Czy potrzebny jest menedżer punktów synchronizacji?	Wymagany produkt (jeden z wymienionych)	Obsługiwane bazy danych na hoście i serwerze System i
Tak	Tak	Serwer DB2 Connect Produkt DB2 Enterprise Server Edition z zastosowaną licencją DB2 Connect	DB2 Universal Database (UDB) for OS/390 and z/OS, wersja 7 DB2 UDB for z/OS, wersja 8 lub nowsza.
Nie	Nie	Produkt DB2 Connect Personal Edition Serwer DB2 Connect Produkt DB2 Enterprise Server Edition z zastosowaną licencją DB2 Connect	DB2 UDB for OS/390 and z/OS, wersja 7 DB2 UDB for z/OS, wersja 8 lub nowsza.

Uwaga: Transakcja rozproszona może zaktualizować dowolną kombinację serwerów baz danych, które są obsługiwane. Na przykład aplikacja użytkownika w ramach pojedynczej transakcji może aktualizować kilka tabel w bazie danych DB2 w systemie operacyjnym Windows, w bazie danych DB2 for OS/390 oraz w bazie danych DB2 for i5/OS.

Konfigurowanie programu DB2 Connect z menedżerem transakcji zgodnym ze standardem XA

W tej sekcji opisano czynności konfiguracyjne umożliwiające korzystanie z serwerów baz danych dla systemów S/390, System i oraz zSeries przy użyciu monitora przetwarzania transakcyjnego (TP).

Monitor przetwarzania transakcyjnego działa, zainstalowano produkt DB2 Connect oraz skonfigurowano i przetestowano połączenie z hostem lub serwerem bazy danych System i.

Aby skonfigurować produkt DB2 Connect do współpracy z serwerami baz danych S/390, System i oraz zSeries przy użyciu monitora przetwarzania transakcyjnego, wykonaj poniższe kroki:

1. Skonfiguruj monitor przetwarzania transakcyjnego tak, aby mógł uzyskać dostęp do produktu DB2 XA Switch. Produkt DB2 XA Switch dostarcza monitorowi przetwarzania transakcyjnego adresy interfejsów API XA produktu DB2 Connect. W każdym monitorze przetwarzania transakcyjnego jest to realizowane w inny sposób.
2. Skonfiguruj monitor przetwarzania transakcyjnego z łańcuchem XA_OPEN z DB2. W każdym monitorze przetwarzania transakcyjnego jest to realizowane w inny sposób. Informacje na temat konfigurowania łańcucha XA OPEN z DB2 w taki sposób, aby mógł być użyty przez monitor przetwarzania transakcyjnego, powinny być zawarte w dokumentacji dostarczonej z danym monitorem przetwarzania transakcyjnego.
3. Jeśli jest to wymagane, zmodyfikuj domyślne parametry konfiguracyjne menedżera punktów synchronizacji (SPM) produktu DB2 Connect. Serwery bazy danych na hoście i serwerze System i (wersja 5 wydanie 3 i starsze) nie obsługują interfejsu XA. Produkt System i, wersja 5 wydanie 4 i nowsze, oferuje pełną obsługę interfejsu XA.

   SPM jest komponentem produktu DB2 Connect, który odwzorowuje protokół zatwierdzania dwufazowego XA na protokół zatwierdzania dwufazowego używany przez hosta i serwery baz danych System i. Domyślnie instancja DB2 ma predefiniowane wartości parametrów konfiguracyjnych menedżera punktów synchronizacji. Parametrem, który ma największe znaczenie, jest parametr konfiguracyjny menedżera bazy danych SPM_NAME. Domyślnie jest to kombinacja pierwszych siedmiu znaków nazwy hosta TCP/IP.
4. W produkcie DB2 for Linux, UNIX, and Windows ustaw zmienną rejestru **DB2COMM** tak, aby używany był protokół TCP/IP i ustaw parametr konfiguracyjny menedżera bazy danych **SVCENAME** na numer portu TCP/IP lub nazwę usługi.

Obsługa luźno powiązanych transakcji w programie DB2 Connect

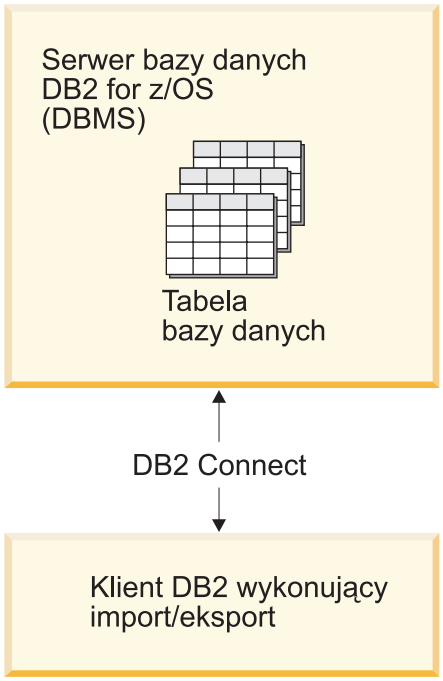
Obsługa luźno powiązanych transakcji w produkcie DB2 Connect jest przeznaczona dla użytkowników, którzy implementują aplikacje rozproszone XA uzyskujące dostęp do produktu DB2 Universal Database (UDB) for i5/OS, wersja 5, wydanie 4, lub nowsze lub DB2 UDB for OS/390 and z/OS, wersja 7 lub nowsza. Pozwala ona różnym rozgałęzieniom tej samej transakcji globalnej współdzielić przestrzeń blokad programu DB2 UDB for OS/390 and z/OS.

Obsługa luźno powiązanych transakcji jest przeznaczona dla aplikacji .NET i COM+.

Opcja ta zmniejsza rozmiar okna, w którym jedno odgałęzienie transakcji rozproszonej napotyka przekroczenie limitu czasu oczekiwania na blokadę lub zakleszczenie spowodowane przez inne odgałęzienie tej samej transakcji globalnej.

Rozdział 8. Przenoszenie danych przy użyciu produktu DB2 Connect

W złożonym środowisku, w którym niezbędne jest przenoszenie danych pomiędzy bazą danych na systemie hostowym i stacją roboczą, można skorzystać z produktu DB2 Connect, stanowiącego gateway dla transmisji danych pomiędzy hostem i stacją roboczą (patrz Rys. 8).



Rysunek 8. Operacje importu i eksportu za pośrednictwem produktu DB2 Connect

Narzędzie eksportujące i importujące produktu DB2 umożliwia przenoszenie danych z hosta lub bazy danych serwera System i do pliku na stacji roboczej DB2 Connect i odwrotnie. Danych tych można następnie użyć w dowolnej innej aplikacji lub systemie zarządzania baz danych, który obsługuje ten format eksportu lub importu. Na przykład dane można wyeksportować z hosta lub bazy danych serwera System i do pliku PC/IXF, a następnie zaimportować do bazy danych DB2 w systemie Windows.

Operacje eksportu i importu można wykonywać z poziomu klienta bazy danych lub stacji roboczej DB2 Connect.

Uwaga:

1. Dane, które mają być importowane lub eksportowane, muszą spełniać warunki dotyczące wielkości i typów danych obowiązujące w obu bazach danych.
2. Aby podnieść wydajność importu, można skorzystać z zapytań złożonych. Aby zgrupować określoną liczbę instrukcji zapytań w jeden blok, należy podać modyfikator złożonego typu pliku w narzędziu importującym. Może to zmniejszyć obciążenie sieci i skrócić czas odpowiedzi.

Operacje eksportu i importu w programie DB2 Connect podlegają następującym ograniczeniom:

© Copyright IBM Corp. 1993, 2010

- Typ pliku musi mieć format PC/IXF.
- Aby można było zaimportować dane na serwer docelowy, należy utworzyć na nim tabelę o atrybutach zgodnych z danymi. Do pobrania atrybutów z tabeli źródłowej można użyć programu narzędziowego db2look. Podczas importu przy użyciu programu DB2 Connect nie można utworzyć tabeli, ponieważ jedyną obsługiwaną instrukcją jest instrukcja INSERT.

Jeśli którykolwiek z tych warunków nie zostanie spełniony, operacja nie powiedzie się i zostanie wyświetlony komunikat o błędzie.

Uwaga: Definicje indeksów nie są zapisywane podczas eksportu ani używane podczas importu.

Jeśli importowane są dane mieszane (kolumny zawierające dane jednobajtowe i dwubajtowe), należy wziąć pod uwagę poniższe ograniczenia:

- W systemach przechowujących dane w formacie EBCDIC (MVS, OS/390, OS/400, VM i VSE) znaki shift-out i shift-in oznaczają początek i koniec danych dwubajtowych. Podczas definiowania szerokości kolumn dla tabel bazy danych należy przewidzieć miejsce dla tych znaków.
- Zalecane są kolumny o zmiennej długości znaków, chyba że dane w kolumnie są zgodne z jednolitym wzorcem.

Przenoszenie danych ze stacji roboczej do serwera hosta

Aby przenieść dane do bazy danych hosta lub serwera System i:

1. Wyeksportuj dane z tabeli DB2 do pliku PC/IXF.
2. Korzystając z opcji INSERT, zaimportuj plik PC/IXF do zgodnej tabeli w bazie danych na serwerze.

Aby przenieść dane z serwera bazy danych na hoście na stację roboczą:

1. Wyeksportuj dane z tabeli bazy danych na hoście do pliku PC/IXF.
2. Zaimportuj plik PC/IXF do tabeli DB2.

Przykład

Poniższy przykład ilustruje sposób przenoszenia danych ze stacji roboczej do bazy danych hosta lub serwera System i.

Wyeksportuj dane w zewnętrznym formacie IXF za pomocą następującej komendy:

```
db2 export to personel.ixf of ixf select * from id_użytkownika.personel
```

Wydaj poniższą komendę, aby ustanowić połączenie DRDA z docelową bazą danych DB2:

```
db2 connect to cbc664 user admin using xxx
```

Jeśli jeszcze nie istnieje, utwórz tabelę docelową w docelowej instancji bazy danych DB2:

```
CREATE TABLE mydb.staff (ID SMALLINT NOT NULL, NAME VARCHAR(9),
       DEPT SMALLINT, JOB CHAR(5), YEARS SMALLINT, SALARY DECIMAL(7,2),
       COMM DECIMAL(7,2))
```

Aby zaimportować dane, użyj następującej komendy:

```
db2 import from personel.ixf of ixf insert into moja_baza_danych.personel
```

Z pliku w formacie IXF zostanie odczytany każdy wiersz, który następnie zostanie wstawiony do tabeli moja_baza_danych.personel za pomocą instrukcji SQL INSERT. Kolejno wstawiane będą następne wiersze, dopóki do tabeli docelowej nie zostaną przeniesione wszystkie dane.

Szczegółowe informacje są dostępne w publikacji dokumentacji technicznej "Moving Data Across the DB2 Family" (IBM Redbooks). Tę publikację dokumentacji technicznej (Redbooks) można znaleźć pod następującym adresem URL: http://www.redbooks.ibm.com/redbooks/SG246905.

Rozdział 9. Odwzorowanie kodów SQLCODE

Różne produkty IBM z rodziny relacyjnych baz danych nie zawsze generują te same kody SQLCODE dla podobnych błędów. Nawet jeśli kod SQLCODE jest taki sam, to może wystąpić z elementami zmiennymi określonymi w różny sposób. Lista elementów jest przekazywana w polu SQLERRMC obszaru komunikacyjnego SQL. Domyślnie program DB2 Connect odwzorowuje kody SQLCODE i tokeny z każdego hosta lub serwera bazy danych System i na odpowiednie kody SQLCODE bazy danych DB2.

Aby wyłączyć odwzorowanie kodów SQLCODE, należy użyć parametru **NOMAP** w łańcuchu parametrów katalogu DCS.

Odwzorowanie kodów SQLCODE można wyłączyć, jeśli aplikacja jest przenoszona bezpośrednio z serwera bazy danych na hoście lub serwerze System i, na przykład z serwera DB2 Universal Database for OS/390 and z/OS. Pozwoli to używać aplikacji bez zmiany kodów SQLCODE, do których się ona odwołuje.

Wyłączanie odwzorowania kodów SQLCODE

Aby wyłączyć odwzorowanie kodów SQLCODE, należy użyć parametru NOMAP w łańcuchu parametrów katalogu DCS.

Odwzorowanie kodów SQLCODE można wyłączyć, jeśli aplikacja jest przenoszona bezpośrednio z serwera bazy danych na hoście lub serwerze System i, na przykład z serwera DB2 Universal Database (UDB) for OS/390 and z/OS. Pozwoli to używać aplikacji bez zmiany kodów SQLCODE, do których się ona odwołuje.

Dostosowanie odwzorowania kodów SQLCODE

Domyślnie program DB2 Connect odwzorowuje kody SQLCODE i tokeny z każdego hosta lub serwera bazy danych System i na odpowiednie kody SQLCODE bazy danych DB2. Poniższe pliki są kopiami domyślnego odwzorowania SQLCODE:

- Plik dcs1dsn.map odwzorowuje kody SQLCODE produktów DB2 Universal Database (UDB) for OS/390 and z/OS oraz DB2 for z/OS.
- Plik dcs1ari.map odwzorowuje kody SQLCODE produktu DB2 UDB for VSE and VM.
- Plik dcs1qsq.map odwzorowuje kody SQLCODE produktów DB2 UDB for iSeries oraz DB2 for i5/OS.

Odwzorowanie nie jest wymagane w przypadku produktu DB2 dla systemu operacyjnego Linux lub UNIX.

1. Aby nadpisać domyślne odwzorowanie kodów SQLCODE lub w sytuacji, gdy używany jest serwer bazy danych na hoście lub serwerze System i, dla którego nie ma odwzorowania kodów SQLCODE (serwer bazy danych firmy innej niż IBM), można skopiować jeden z tych plików i użyć go jako podstawy dla nowego pliku odwzorowań kodów SQLCODE. Skopiowanie pliku, a nie bezpośrednia jego edycja, daje pewność, że w razie konieczności można odwołać się do oryginalnego odwzorowania SQLCODE.
2. W łańcuchu parametrów katalogu DCS należy podać nazwę pliku nowego pliku odwzorowania SQLCODE.

© Copyright IBM Corp. 1993, 2010

3. Każdy plik odwzorowania jest plikiem ASCII utworzonym i modyfikowanym przy użyciu edytora ASCII. W czasie instalacji plik jest zapisywany w katalogu map ścieżki instalacji.

   Plik ten może zawierać następujące specjalne typy wierszy:

   && Logiczny początek pliku. Wszystkie wiersze znajdujące się przed pierwszym wystąpieniem znaków && są uważane za komentarz i są pomijane. Jeśli po znakach && nie występują żadne znaki, to odwzorowanie SQLCODE nie jest wykonywane. Można również wyłączyć odwzorowanie SQLCODE, używając parametru NOMAP w sposób opisany wcześniej.

   ***** Jeśli jest pierwszym znakiem w wierszu, oznacza komentarz.

   W Jeśli jest jedynym znakiem w wierszu, oznacza flagi ostrzeżeń, które powinny być ponownie odwzorowane. Domyślnie przekazywane są oryginalne flagi ostrzeżeń. 'W' musi być wielką literą.

   Wszystkie pozostałe wiersze po znakach && muszą być puste albo muszą być instrukcjami odwzorowania w następującej postaci:

   ```
   kod_wejściowy [, kod_wyjściowy [, lista_tokenów]]
   ```

   Parametr *kod_wejściowy* może reprezentować:

   kod_sql
   Kod SQLCODE z serwera bazy danych na hoście lub serwerze System i.

   U Wszystkie niezdefiniowane kody SQLCODE o wartości ujemnej (niewymienione w tym pliku) są odwzorowane na określony *kod_wyjściowy*. Jeśli w tym wierszu nie podano *kodu_wyjściowego*, używany jest oryginalny kod SQLCODE. Ten znak musi być pisany wielką literą.

   P Wszystkie niezdefiniowane kody SQLCODE o wartości dodatniej (niewymienione w tym pliku) są odwzorowane na określony *kod_wyjściowy*. Jeśli w tym wierszu nie podano *kodu_wyjściowego*, używany jest oryginalny kod SQLCODE. Ten znak musi być pisany wielką literą.

   cc*nn* Kod SQLSTATE z serwera bazy danych na hoście lub serwerze System i. *nn* może przyjąć jedną z następujących wartości:

   00 Niekwalifikowane pomyślne zakończenie

   01 Ostrzeżenie

   02 Brak danych

   21 Naruszenie liczności

   22 Wyjątek dotyczący danych

   23 Naruszenie ograniczenia

   24 Niepoprawny stan kursora

   26 Niepoprawny identyfikator instrukcji SQL

   40 Wycofanie zmian transakcji

   42 Naruszenie dostępu

   51 Niepoprawny stan aplikacji

   55 Obiekt nie spełnia warunków wstępnych

   56 Inne błędy języka SQL lub produktu

   57 Zasób niedostępny lub interwencja operatora

   58 Błąd systemowy.

Podany *kod_wyjściowy* jest używany dla wszystkich kodów SQLCODE z danym kodem klasy, które nie są jawnie podane w pliku odwzorowania. Jeśli w tym wierszu nie podano *kodu_wyjściowego*, oryginalny SQLCODE jest odwzorowany na siebie bez nadpisywania znaczników.

Znaki **cc** muszą być pisane małymi literami.

Jeśli w pliku odwzorowania ten sam *kod_wejściowy* występuje więcej niż raz, używane jest jego pierwsze wystąpienie. Parametr *kod_wyjściowy* reprezentuje wyjściowy kod SQLCODE. Jeśli nie podano żadnej wartości, używany jest oryginalny kod SQLCODE.

Jeśli podawany jest kod wyjściowy, można również podać jeden z następujących parametrów:

(s) Kod wejściowy SQLCODE i identyfikator produktu (ARI, DSN lub QSQ) będą umieszczone w polu znacznika komunikatu obszaru komunikacyjnego SQL.

Oryginalny kod SQLCODE jest zwracany jako jedyny znacznik. Opcję tę zaprojektowano do obsługi niezdefiniowanych kodów SQLCODE, z wyjątkiem wartości +965 i -969. Jeśli *kodem_wyjściowym* jest +965 lub -969, lista znaczników zwracana w polu SQLERRMC obszaru komunikacyjnego SQL zawiera oryginalny kod SQLCODE, po którym występuje identyfikator produktu, a następnie oryginalna lista znaczników.

Znak **s** musi być pisany małą literą.

(*lista-tokenów***)**

Lista znaczników oddzielonych przecinkami. Aby pominąć dany znacznik, należy wpisać przecinek. Na przykład zapis (,*t2*,,*t4*) oznacza, że pierwszy i trzeci znacznik wyjściowy są pomijane.

Każdy token ma postać liczby (*n*), opcjonalnie poprzedzonej przez znak **c** oraz opcjonalnie poprzedzającej znaki **c** lub **i**. Są one interpretowane w następujący sposób:

c Typ danych znacznika z tą pozycją to CHAR (domyślnie). Jeśli znak **c** występuje przed *n*, to odwołuje się on do znacznika wejściowego, jeśli występuje po znaku *n*, to odwołuje się do znacznika wyjściowego. Znak **c** musi być pisany małą literą.

i Typ danych znacznika z tą pozycją to INTEGER. Jeśli znak **i** występuje po *n*, to odwołuje się do znacznika wyjściowego. Znak **i** nie powinien występować przed *n*, ponieważ produkty serwerów baz danych IBM na hoście lub serwerze System i obsługują tylko elementy zmienne typu CHAR. Znak **i** musi być pisany małą literą.

n Liczba lub liczby wskazujące, które elementy zmienne z serwera bazy danych na hoście lub serwerze System i są używane. Są one uporządkowane w takiej kolejności, w jakiej mają być umieszczane w wyjściowym obszarze komunikacyjnym SQL. Liczba wskazuje element serwera bazy danych na hoście lub serwerze System i; wzajemne ustawienie oznacza kolejność, w jakiej elementy zostaną umieszczone w obszarze SQLCA.

Na przykład serwer bazy danych na hoście lub serwerze System i może zwrócić dwa elementy zmienne: 1 i 2. Aby element 2 pojawił się w wyjściowym obszarze SQLCA przed elementem 1, należy określić zapis (2,1).

Wiele numerów elementów można połączyć kropkami, aby utworzyły one jeden znacznik wyjściowy typu CHAR.

Do oddzielania znaczników wyjściowych używane są przecinki. Jeśli przed przecinkiem nie podano znacznika, to dla tej pozycji nie jest dołączany znacznik wyjściowy w obszarze komunikacyjnym SQL. Znaczniki występujące w wyjściowym obszarze komunikacyjnym SQL po ostatnim podanym znaczniku są odwzorowane na puste znaczniki.

Rys. 9 przedstawia przykładowy plik odwzorowania SQLCODE.

```
&&
  -007    ,   -007   ,   (1)
  -010
  -060    ,   -171   ,   (2)
...
  -204    ,   -204   ,   (c1.2c)
...
  -633    ,   -206   ,   (,c1i)

  -30021  ,   -30021 ,   (c1c,c2c)

  cc00    ,   +000
...
  U       ,   -969   ,   (s)
  P       ,   +965   ,   (s)
```

Rysunek 9. Plik odwzorowania SQLCODE.

Poniższe opisy odpowiadają pasującym numerom linii na poprzednim rysunku:

1. Kod SQLCODE jest odwzorowany z -007 na -007. Pierwszy element wejściowy odebrany z serwera bazy danych na hoście lub serwerze System i jest używany jako pierwszy element wyjściowy i domyślnie jest on typu CHAR. Inne elementy nie są przekazywane.
2. Kod SQLCODE jest odwzorowany z -010 na -010 (nie podano kodu wyjściowego SQLCODE). Brak znaczników umieszczanych w wyjściowym obszarze komunikacyjnym SQL.
3. Kod SQLCODE jest odwzorowany z -060 na -171. Pierwszy element wejściowy odebrany z serwera bazy danych na hoście lub serwerze System i jest usuwany. Drugi jest używany jako pierwszy element w wyjściowym obszarze komunikacyjnym SQL, a jego typ danych to CHAR. Brak drugiego znacznika w wyjściowym obszarze komunikacyjnym SQL.
4. Kod SQLCODE jest odwzorowany z -204 na -204. Elementy wejściowe, pierwszy i drugi, odebrane z serwera bazy danych na hoście lub serwerze System i są typu CHAR. Te dwa elementy wejściowe są łączone do postaci jednego elementu wyjściowego o typie CHAR, który będzie pierwszym znacznikiem wyjściowym w obszarze komunikacyjnym SQL.
5. Kod SQLCODE jest odwzorowany z -633 na -206. Pierwszy element wejściowy odebrany z serwera bazy danych na hoście lub serwerze System i jest typu CHAR. Jest on przekształcany do typu INTEGER i jest używany jako drugi znacznik w wyjściowym obszarze komunikacyjnym SQL. Pierwszy znacznik w wyjściowym obszarze komunikacyjnym SQL jest pusty, co oznaczono przecinkiem.
6. Kod SQLCODE jest odwzorowany z -30021 na -30021. Elementy wejściowe, pierwszy i drugi, odebrane z serwera bazy danych na hoście lub serwerze System i, są typu CHAR i używane są jako pierwszy i drugi element w wyjściowym obszarze SQLCA.
7. Wszystkie kody SQLCODE w obszarach komunikacyjnych SQL z SQLSTATE z klasy 00 będą odwzorowane na kod SQLCODE +000.
8. Wszystkie niezdefiniowane kody SQLCODE są odwzorowane na -969. Opcja ta powinna być używana, tylko jeśli wszystkie kody, które mogą zostać odwzorowane, są podane,

łącznie z kodami identycznymi, niewymagającymi odwzorowania. Opcja **(s)** oznacza, że lista znaczników, która ma być zwrócona w polu SQLERRMC obszaru komunikacyjnego SQL, zawiera oryginalny kod SQLCODE. Po nim występuje produkt, w którym wystąpił błąd, a następnie oryginalna lista znaczników. Jeśli nie podano pozycji **U**, wszystkie kody nieobecne na liście są przesyłane bez odwzorowania.

9. Wszystkie niezdefiniowane kody SQLCODE o znaku dodatnim są odwzorowane na +965. Opcja ta powinna być używana, tylko jeśli wszystkie kody, które mogą zostać odwzorowane, są podane, łącznie z kodami identycznymi, niewymagającymi odwzorowania. Opcja **(s)** oznacza, że lista znaczników, która ma być zwrócona w polu SQLERRMC obszaru komunikacyjnego SQL, zawiera oryginalny kod SQLCODE. Po nim występuje produkt, w którym wystąpiło ostrzeżenie, a następnie oryginalna lista znaczników. Jeśli nie podano pozycji **P**, wszystkie kody o znaku dodatnim na liście są przesyłane bez odwzorowania.

Rozdział 10. Monitorowanie systemów baz danych i program DB2 Connect

Przedstawiono tu kilka sposobów monitorowania połączeń i wydajności w środowiskach korzystających z programu DB2 Connect. Typ monitorowania zależny jest od systemu operacyjnego.

Monitorowanie połączeń klientów zdalnych

W ramach produktu DB2 Connect, takiego jak DB2 Connect Enterprise Edition, można skorzystać z narzędzia monitor systemu baz danych do monitorowania połączeń zdalnych klientów. Do monitorowania klientów lokalnych względem serwera DB2 Connect, uruchomionych na samym serwerze, konieczne jest ustawienie następującej zmiennej:

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

Na przykład, gdy w systemie hosta lub w systemie System i wystąpi błąd, administrator systemu może określić, czy problem dotyczy stacji roboczej DB2 Connect. Monitor systemu baz danych współpracuje z następującymi elementami:

- Elementem korelacji (CRRTKN) DRDA w przypadku konwersacji niechronionych.
- identyfikatorem jednostki pracy (UOWID) w wypadku połączeń dwufazowych zabezpieczonych przez menedżera punktów synchronizacji DRDA-3 (używanego w połączeniach TCP/IP)
- identyfikatorem połączenia DB2 Connect (identyfikatorem aplikacji)

Powyższe informacje umożliwiają określenie, które połączenie DB2 Connect spowodowało wystąpienie problemu, co umożliwia administratorowi systemu oddzielenie indywidualnej aplikacji klienta od systemu bez wpływu na innych klientów korzystających z połączenia DB2 Connect.

Wyświetlanie statusu parametrów monitorowania

Aby wyświetlić status parametrów monitorowania, należy użyć komendy db2 get monitor switches.

Korzystanie z Monitora wydajności w systemie Windows

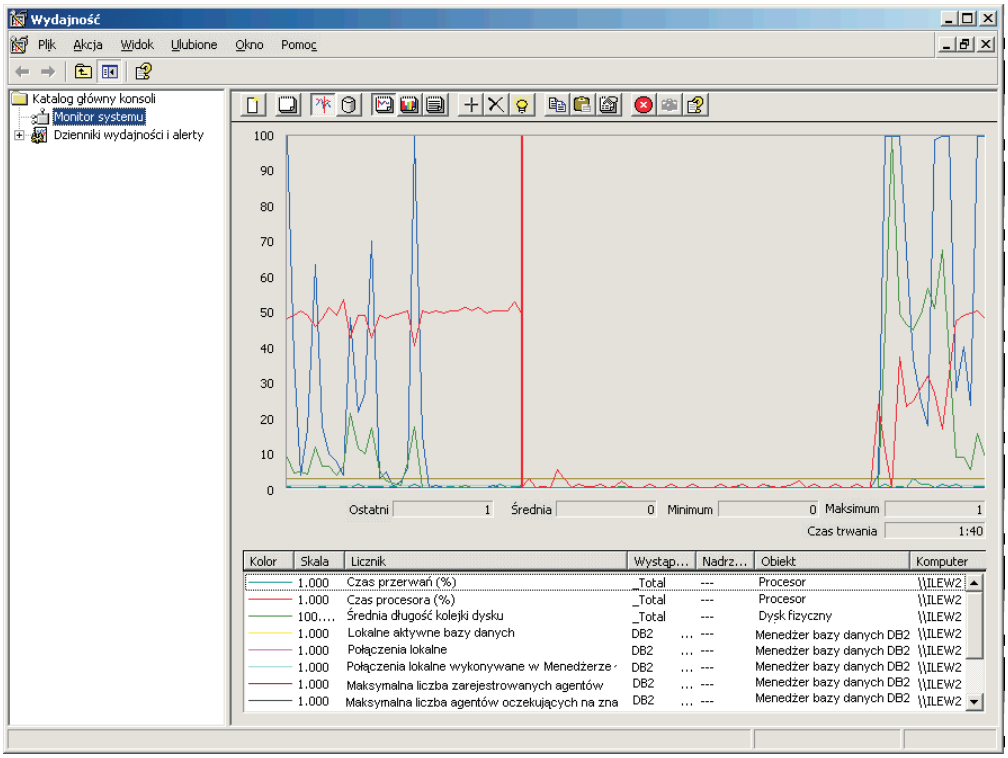
Systemy operacyjne Windows zawierają użyteczne narzędzie do monitorowania wydajności aplikacji DB2. Monitor wydajności, który jest jednym z narzędzi administracyjnych systemu Windows, wyświetla informacje o wydajności systemu w postaci graficznej. Do monitorowania można wybrać różne systemy, bazy danych i pozycje związane z komunikacją i odwzorować je razem w postaci graficznej.

Na przykład raporty dostępne za pomocą komendy GET SNAPSHOT FOR ALL DCS DATABASES lub komendy GET SNAPSHOT FOR ALL DCS APPLICATIONS mogą być wyświetlone w sposób graficzny w czasie rzeczywistym za pomocą monitora i porównane bezpośrednio z takimi wartościami, jak na przykład wykorzystanie czasu pracy procesora. Można bezpośrednio porównywać wpływ różnych ustawień na bazę danych lub wydajność komunikacji. Ustawienia specjalnych konfiguracji można zapisywać w plikach PMC, które później można pobierać.

Na rysunku przedstawiono kilka parametrów DB2 w sposób graficzny i porównano je z wykorzystaniem czasu pracy procesora. Kolekcja wartości prezentowanych na wykresach

© Copyright IBM Corp. 1993, 2010

została zapisana w pliku db2chart.pmc. Można zapisać dowolną liczbę plików PMC i w każdym z nich uwzględnić inny przegląd wydajności systemu.



Rysunek 10. Monitor wydajności

Aby włączyć monitorowanie lokalnych aplikacji, niezbędne jest wyłączenie zmiennej środowiskowej DB2CONNECT_IN_APP_PROCESS.

Korzystanie z komend GET SNAPSHOT

Monitor DB2 na bieżąco rejestruje ważne informacje systemowe. W dowolnym momencie można uzyskać podsumowanie statusu systemu, wprowadzając komendę GET SNAPSHOT. Użytkownik może uzyskać obrazy stanu z monitora, jeśli ma uprawnienie SYSMAINT, SYSCTRL lub SYSADM do instancji menedżer bazy danych, którą chce monitorować.

Istnieje pięć komend tworzenia obrazów, które są przydatne podczas monitorowania informacji DCS. Są to:

- GET SNAPSHOT FOR ALL DCS DATABASES
- GET SNAPSHOT FOR ALL DCS APPLICATIONS
- GET SNAPSHOT FOR DCS APPLICATION ...
- GET SNAPSHOT FOR DCS DATABASE ON alias_bazy_danych
- GET SNAPSHOT FOR DCS APPLICATIONS ON alias_bazy_danych

Każda komenda tworzenia obrazu stanu powoduje utworzenie szczegółowego raportu dotyczącego żądanego obszaru.

Na przykład wprowadzenie komendy GET SNAPSHOT FOR DCS DATABASE ON DCSDB spowoduje utworzenie następującego raportu:

```
                Obraz bazy danych DCS

Nazwa bazy danych DCS                          = DCSDB
```

```
Nazwa bazy danych hosta                         = GILROY
Datownik pierwszego połączenia z bazą danych    = 12-15-2001 10:28:24.596495
Najnowszy czas do połączenia                    = 0.950561
Najnowszy czas trwania połączenia               = 0.000000
Czas odpowiedzi hosta (s.ms)                    = 0.000000
Datownik ostatniego resetowania                 =
Liczba wydanych instrukcji SQL                  = 2
Wydane instrukcje zatwierdzenia                 = 1
Wydane instrukcje wycofania zmian               = 0
Operacje instrukcji zakończone niepowodzeniem   = 0
Całkowita liczba połączeń przez gateway         = 1
Aktualna liczba połączeń gatewaya               = 1
Połączenia gatewaya oczekujące na odpowiedź hosta = 0
Połączenia gatewaya oczekujące na żądanie klienta = 1
Błędy komunikacyjne gatewaya do hosta           = 0
Datownik ostatniego błędu komunikacyjnego       = Brak
Wskaźnik wysokiego poziomu dla połączeń gatewaya  = 1
Wybrane wiersze                                 = 0
Wysłano bajtów wychodzących                     = 140
Otrzymano bajtów wychodzących                   = 103
```

Raport ten dostarcza informacji na temat połączeń bazy danych, wydajności, błędów i przepustowości żądań SQL. Obrazy stanu z monitora DB2 mogą być w rzeczywistości o wiele bardziej szczegółowe. Na przykład, jeśli zostanie wydana komenda GET SNAPSHOT FOR ALL DCS APPLICATIONS, uzyskany raport będzie przypominał poniższy:

```
          Obraz aplikacji DCS

Identyfikator aplikacji klienckiej         = 09150F74.B6A4.991215152824
  Numer kolejny                            = 0001
  ID autoryzowanego użytkownika            = SMITH
  Nazwa aplikacji                          = db2bp
  Uchwyt aplikacji                         = 1
  Status aplikacji                         = oczekiwanie na żądanie
  Czas zmiany statusu                      = 12-15-2001 10:29:06.707086
  Węzeł kliencki                           = sys143
  Poziom wersji klienta                    = SQL06010
  Platforma klienta                        = AIX
  Protokół klienta                         = TCP/IP
  Strona kodowa klienta                    = 850
  ID procesu aplikacji klienta             = 49074
  Identyfikator logowania klienta          = smith
  ID aplikacji hosta                       = G9150F74.B6A5.991215152825
  Numer kolejny                            = 0000
  Alias bazy danych na gatewayu            = MVSDB
  Nazwa bazy danych DCS                    = DCSDB
  Nazwa bazy danych hosta                  = GILROY
  Poziom wersji hosta                      = DSN05012
  CCSID hosta                              = 500

Adres dla komunikacji wychodzącej             = 9.21.21.92 5021
Protokół dla komunikacji wychodzącej          = TCP/IP
Adres dla komunikacji przychodzącej           = 9.21.15.116 46756
Datownik pierwszego połączenia z bazą danych  = 12-15-2001 10:28:24.596495
Czas odpowiedzi hosta (s.ms)                  = 0.000000
Czas przetwarzania przez gateway              = 0.000000
Datownik ostatniego resetowania               =
Wybrane wiersze                               = 0
Liczba wydanych instrukcji SQL                = 2
Operacje instrukcji zakończone niepowodzeniem = 0
Zatwierdzone instrukcje                       = 1
Liczba instrukcji wycofania zmian             = 0
Otrzymano bajtów przychodzących               = 404
Wysłano bajtów wychodzących                   = 140
Otrzymano bajtów wychodzących                 = 103
Wysłano bajtów przychodzących                 = 287
Liczba otwartych kursorów                     = 0
```

```
Czas bezczynności aplikacji                   = 1 minuta i 32 sekundy

Status wykonania jednostki pracy              =
Datownik zatrzymania poprzedniej jednostki    = 12-15-2001 10:28:25.592631
Datownik uruchomienia jednostki pracy         = 12-15-2001 10:29:06.142790
Datownik zatrzymania jednostki pracy          =
Czas ostatniej zakończonej jednostki pracy (s.ms)= 0.034396

Ostatnia operacja                             = Wykonaj natychmiast
Datownik uruchomienia ostatniej operacji      = 12-15-2001 10:29:06.142790
Datownik zatrzymania ostatniej operacji       = 12-15-2001 10:29:06.707053

Instrukcja                                    = Wykonaj natychmiast
Numer sekcji                                  = 203
Autor aplikacji                               = NULLID
Nazwa pakietu                                 = SQLC2C07
Przewidywany koszt kompilatora SQL (w jednostkach timeron) = 0
Szacunkowa liczność kompilatora SQL           = 0
Datownik uruchomienia instrukcji              = 12-15-2001 10:29:06.142790
Datownik zatrzymania instrukcji               = 12-15-2001 10:29:06.707053
Czas odpowiedzi hosta (s.ms)                  = 1.101612
Czas ostatniej zakończonej jednostki pracy (s.ms)= 0.564263
Wiersze pobrane                                  = 0
Czas przetwarzania przez gateway                 = 0.013367
Otrzymano bajtów przychodzących dla instrukcji   = 220
Wysłano bajtów wychodzących dla instrukcji       = 130
Otrzymano bajtów wychodzących dla instrukcji     = 49
Wysłano bajtów przychodzących dla instrukcji     = 27
Tekst instrukcji SQL:
create table t12 (col1 int, col2 char)
```

Status aplikacji DCS

Monitor systemu udostępnia trzy formy komendy LIST DCS APPLICATIONS, jak poniżej:

- LIST DCS APPLICATIONS
- LIST DCS APPLICATIONS SHOW DETAIL
- LIST DCS APPLICATIONS EXTENDED

W informacji otrzymanej w wyniku uruchomienia tych komend format identyfikatora aplikacji hosta i identyfikatora aplikacji klienta może się różnić w zależności od wersji hosta lub bazy danych System i oraz poziomu obsługi protokołu TCP/IP.

Tabela 11. Format identyfikatora aplikacji według wersji hosta i poziomu obsługi protokołu TCP/IP

Scenariusz	Format identyfikatora aplikacji
Klienty uzyskujące dostęp do serwerów danych z poziomem obsługi menedżera RDB mniejszym niż 7	G91A0D3A.P8BC.060306212019
Klienty uzyskujące dostęp do serwerów danych z poziomem obsługi Menedżera RDB 8 lub większym poprzez protokół TCP/IP, wersja 4	9.26.13.61.65289.060306213816

Tabela 11. Format identyfikatora aplikacji według wersji hosta i poziomu obsługi protokołu TCP/IP (kontynuacja)

Scenariusz	Format identyfikatora aplikacji
Klienty uzyskujące dostęp do serwerów danych z poziomem obsługi Menedżera RDB 8 lub większym poprzez protokół TCP/IP, wersja 6	2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741

LIST DCS APPLICATIONS

Aby przeglądać informacje dostarczone przez monitor na poziomie aplikacji, należy wydać komendę DB2 LIST DCS APPLICATIONS.

Dla połączenia TCP/IP (pomiędzy produktami DB2 Connect i DB2 Universal Database for z/OS and OS/390) zwraca ona poniższą informację:

```
Id      Nazwa            Uchwyt   Identyfikator
autora  aplikacji        aplik.   aplikacji hosta
------- ---------------- ------ --------------------------------------------------
NEWTON  db2cli.exe     7        G91A0D3A.P8BC.060306212019
NEWTON  db2cli.exe     25       9.26.13.61.65289.060306213816
NEWTON  db2cli.exe     20       2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741
```

Identyfikator autoryzowanego użytkownika
Jest to ID autoryzowanego użytkownika, który został użyty do zalogowania się na serwerze bazy danych na hoście lub serwerze System i. Określa on, kto uruchomił aplikację.

Nazwa aplikacji
Nazwa aplikacji uruchomionej na kliencie podana programowi DB2 Connect. Dostępnych jest tylko 20 pierwszych bajtów po ostatnim separatorze ścieżki.

Uchwyt aplikacji
Agent wykonywany na stacji roboczej DB2 Connect. Elementu tego można użyć do połączenia informacji monitora monitor systemu baz danych z innymi informacjami diagnostycznymi. Podczas używania komendy FORCE USERS lub funkcji API wymagany jest również identyfikator agenta.

identyfikator aplikacji hosta
Identyfikator ten może być:

- Elementem korelacji (CRRTKN) DRDA w przypadku konwersacji niechronionych.
- Identyfikatorem jednostki pracy (UOWID) dla dwufazowych połączeń zabezpieczonych przez DRDA-3 Menedżera punktów synchronizacji (używanych przez połączenia TCP/IP).

Ten unikalny identyfikator jest generowany, gdy aplikacja łączy się z serwerem bazy danych na hoście lub serwerze System i. Elementu tego można używać w połączeniu z identyfikatorem aplikacji w celu skorelowania informacji dotyczących aplikacji po stronie klienta i po stronie serwera.

LIST DCS APPLICATIONS SHOW DETAIL

Jeśli zostanie podany format komendy DB2 LIST DCS APPLICATIONS SHOW DETAIL, zostaną wyświetlone dodatkowe informacje:

```
Id autoryz. uż.               Nazwa aplikacji      Uchwyt     Id aplikacji klienta
                                                   aplikacji
----------------------------- -------------------- ---------- --------------------------------------------------
NEWTON                        db2cli.exe           37         2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224


Lp.   Alias BD  Węzeł    Wersja   Str. kod.  Id aplikacji hosta
      klienta   klienta  klienta  klienta
----- --------- -------- -------- ---------- -------------------------
00001 MDB       SAYYID   SQL09000 1252       G91A0D3A.P982.060306214231


Lp.   Nazwa BD hosta       Wersja
                           hosta
----- -------------------- --------
00001 MEXICO               DSN08015
```

Identyfikator aplikacji klienta
Jednoznacznie identyfikuje aplikację podłączoną do stacji roboczej DB2 Connect. Istnieją różne formaty identyfikatora aplikacji, które zależą od protokołu komunikacyjnego używanego do komunikacji między klientem i stacją roboczą DB2 Connect.

Wartość ta pozwala skorelować połączenia z klientów do stacji roboczej DB2 Connect i ze stacji roboczej DB2 Connect do serwera bazy danych na hoście lub serwerze System i.

Numer kolejny (Nr)
Numer kolejny klienta jest numerem kolejnym transakcji. Został on utworzony, aby ułatwić skorelowanie transakcji rozproszonej po różnych systemach.

Alias bazy danych klienta
Alias bazy danych jest dostarczany w celu nawiązania przez aplikację połączenia z bazą danych. Elementu tego można używać do identyfikacji rzeczywistej bazy danych, do której aplikacja ma dostęp. Odwzorowanie między tą nazwą oraz nazwą bazy danych można wykonać, używając katalogów bazy danych w węźle klienta i węźle serwera menedżera menedżer bazy danych.

NNAME klienta (węzeł)
Identyfikuje węzeł, w którym wykonywana jest aplikacja klienta. Pole może zawierać różne informacje w zależności od protokołu używanego przez klienta. Dla klienta podłączonego przez protokół TCP/IP jest to nazwa hosta.

Identyfikator produktu klienta (klient)
Produkt oraz wersja uruchomiona na kliencie. Identyfikatorem produktu klienta będą:

- SQL07010 dla produktów DB2 Universal Database i DB2 Connect wersja 7.1 oraz ich klientów.
- SQL08010 dla produktów DB2 Universal Database i DB2 Connect wersja 8.1 oraz ich klientów.
- SQL08020 dla produktów DB2 Universal Database i DB2 Connect wersja 8.2 oraz ich klientów.
- SQL09120 dla produktów DB2 i DB2 Connect w wersji 9.1 oraz ich klientów.

Identyfikator strony kodowej
Identyfikator strony kodowej w węźle, w którym została uruchomiona monitorowana aplikacja.

Informacje te można wykorzystać do konwersji danych między stroną kodową aplikacji a stroną kodową bazy danych (lub dla baz danych na hoście lub serwerze System i, identyfikatora CCSID serwera bazy danych na hoście lub serwerze System i).

Jeśli strona kodowa aplikacji różni się od tej, z którą został uruchomiony monitor systemu baz danych, ten element strony kodowej może pomóc w ręcznym przekształceniu danych, które zostały przekazane z aplikacji i wyświetlone przez monitor systemu baz danych. Na przykład może on być pomocny w konwersji nazwy aplikacji.

Kolejny numer wychodzący
Określa kolejny numer wychodzący. Jest on używany do korelowania transakcji w różnych systemach.

Nazwa bazy danych hosta
Rzeczywista nazwa bazy danych, do której została podłączona aplikacja. W katalogu DCS jest to *nazwa docelowej bazy danych*.

Identyfikator produktu hosta
Uruchomiony na serwerze produkt i jego wersja. Ma on format *PPPVVRRM*, gdzie:

PPP identyfikuje hosta lub serwer bazy danych System i (na przykład DSN for DB2 Universal Database for z/OS and OS/390, ARI for DB2 Server for VSE & VM lub QSQ for DB2 for i5/OS);

VV reprezentuje dwucyfrowy numer wersji, (na przykład 08);

RR reprezentuje dwucyfrowy numer wydania, (na przykład 01);

M reprezentuje jednocyfrowy numer modyfikacji (0-9 lub A-Z).

LIST DCS APPLICATIONS EXTENDED

Komendy LIST DCS APPLICATIONS można używać z opcją EXTENDED, aby wygenerować raport rozszerzony. Raport rozszerzony zawiera wszystkie pola, które są wyświetlane, gdy w komendzie określona jest opcja SHOW DETAIL, a ponadto dziewięć nowych pól:

- Status aplikacji DCS
- Godzina zmiany statusu
- Platforma klienta
- Protokół klienta
- Identyfikator kodowanego zestawu znaków CCSID hosta
- ID logowania klienta
- ID procesu aplikacji klienta
- Alias bazy danych w gatewayu
- Nazwa bazy danych DCS

Podczas gdy istniejące opcje komendy wyświetlają pola poziomo, jeden wiersz na aplikację, nowa opcja wyświetla je pionowo, jedno pole na wiersz.

Oto nowa składnia komendy:

```
LIST DCS APPLICATIONS [SHOW DETAIL | EXTENDED ]
```

Nowe informacje wyjściowe, gdy używana jest nowa opcja EXTENDED:

```
                Lista aplikacji DCS - Raport rozszerzony

ID aplikacji klienta = 2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224
Numer kolejny                            = 00001
ID autoryzowanego użytkownika            = NEWTON
Zaufany ID autoryzowanego użytkownika    =
Nazwa aplikacji                          = db2cli.exe
Uchwyt aplikacji                         = 37
Status aplikacji                         = oczekiwanie na żądanie
Czas zmiany statusu                      = nieodczytany
Węzeł klienta                            = SAYYID
Poziom wersji klienta                    = SQL09000
Platforma klienta                        = NT
Protokół klienta                         = TCP/IP
Strona kodowa klienta                    = 1252
ID procesu aplikacji klienta             = 1192
Identyfikator logowania klienta          = ISAYYID
ID aplikacji hosta                       = G91A0D3A.P982.060306214231
Numer kolejny                            = 00001
Alias bazy danych w gatewayu             = MDB
Nazwa bazy danych DCS                    = MDB
Nazwa bazy danych hosta                  = MEXICO
Poziom wersji hosta                      = DSN08015
CCSID hosta                              = 1208
```

Pole statusu aplikacji zawiera jedną z następujących trzech wartości:

1. łączenie w toku - wychodzące. Oznacza, że żądanie połączenia z hostem lub bazą danych System i zostało wysłane, a produkt DB2 Connect oczekuje na nawiązanie połączenia.
2. oczekiwanie na żądanie. Oznacza, że połączenie z bazą danych na hoście lub serwerze System i zostało nawiązane i program DB2 Connect oczekuje na instrukcję SQL z aplikacji klienta.
3. oczekiwanie na odpowiedź. Oznacza, że instrukcja SQL została wysłana do bazy danych na hoście lub serwerze System i.

Również godzina zmiany statusu zostanie wyświetlona w raporcie tylko wtedy, gdy podczas przetwarzania zostanie włączony monitor systemu UOW. W przeciwnym wypadku zostanie wyświetlony komunikat "Nie zebrane".

Monitor poprawności i alerty

Monitor poprawności w produkcie DB2 for z/OS sprawdza okresowo strategie konserwacji obiektów. Jeśli monitor poprawności stwierdzi potrzebę konserwacji obiektu, tworzone są alerty poprawności. Istnieje możliwość przeglądania, uruchamiania i zapisywania działań, które mają zostać podjęte w wyniku alertu poprawności.

Przegląd monitora poprawności produktu DB2 for z/OS

W systemach z/OS monitor produktu DB2 for z/OS jest uruchamiany jako zadanie dla każdego podsystemu DB2, który ma być monitorowany, oraz na dedykowanym elemencie grupy współużytkowania danych.

Monitor poprawności DB2 for z/OS wywołuje wartościowanie strategii konserwacji obiektów w zaplanowanym czasie i w określonych interwałach, zgodnie z definicją zawartą w strategii. Strategie konserwacji obiektów tworzy się za pomocą kreatora tworzenia strategii konserwacji obiektów z Centrum sterowania DB2. Każde wartościowanie strategii polega na porównywaniu kryteriów proponowanej konserwacji z progami ustawionymi w strategii konserwacji obiektów w celu określenia potrzeby konserwacji obiektów, czyli stwierdzenia,

czy są wymagane instrukcje COPY, REORG, RUNSTATS, STOSPACE, ALTER TABLESPACE lub ALTER INDEX, oraz w celu zidentyfikowania stanów zastrzeżonych, takich jak CHKP, dla obszaru tabel, indeksu i obiektów grupy pamięci. Gdy podczas wartościowania strategii jakieś obiekty zostaną zidentyfikowane jako w stanie alertu, kontakty alertów poprawności strategii są powiadamiane przy użyciu adresów poczty elektronicznej lub numerów pagerów. Listę kontaktów alertów poprawności dla każdego podsystemu DB2 definiuje się w Centrum sterowania. Tam też można nią zarządzać.

Monitor poprawności w chwili uruchomienia tworzy początkowy obraz stanu harmonogramu wartościowania dla strategii, za pomocą którego monitor poprawności wyznacza, kiedy uaktywnić wartościowanie strategii. Ten obraz stanu harmonogramu jest odświeżany w czasie odświeżania (określonym przy starcie monitora poprawności) lub w chwili otrzymania komendy odświeżania. Każda zmiana harmonogramu wartościowania strategii jest wykrywana przez monitor poprawności w chwili odświeżania harmonogramu.

Monitor poprawności uruchamia się i zatrzymuje z konsoli za pomocą komend systemu MVS, odpowiednio: START i STOP.

Przykładowa procedura wpisana do katalogu (DSNHMONP), która uruchamia monitor poprawności DB2, i przykładowa procedura wpisana do katalogu (DSNHMONA), która uruchamia wiele monitorów poprawności DB2 w systemie MVS lub klastrze Parallel Sysplex, są umieszczane w bibliotece procedur przez zadanie instalacji DSNTIJHM.

Widoki, tabele, zestawy danych, procedury wpisane do katalogu, procedury składowane, funkcje zdefiniowane przez użytkownika i tabela wynikowa, które są używane przez monitor poprawności db2 lub wymienione niżej zadania pokrewne, są tworzone i instalowane przez zadania instalacji DSNTIJCC i DSNTIJHM. Zadania DSNTIJCC i DSNTIJHM są dostarczane z obiektami FMID JDB771D i JDB881D.

Dziennik wartościowania strategii

Zdarzenia wartościowania strategii wywołane przez monitor poprawności DB2 są rejestrowane w tabeli DSNACC.HM_EVAL_LOG. Wpis jest rejestrowany w chwili, gdy wartościowanie strategii się rozpoczyna i gdy się kończy. Wpisy dziennika są przechowywane przez 7 dni, a potem są z tabeli usuwane. Do wyświetlania wszystkich strategii, których ostatnia iteracja wartościowania nie była pomyślna, można używać widoku DB2 DSNACC.HM_ALERT_PO_EV, który został utworzony dla tej tabeli przez zadanie instalacji DSNTIJCC.

Uruchamianie, zatrzymywanie i odświeżanie monitora poprawności produktu DB2 for z/OS

W systemie z/OS monitor poprawności produktu DB2 for z/OS jest uruchamiany jako zadanie dla każdego monitorowanego podsystemu DB2 lub na dedykowanym elemencie grupy współużytkowania danych.

- Aby uruchomić monitor poprawności DB2, należy wprowadzić następującą komendę START systemu MVS:

  ```
  S nazwa_elementu,DB2SSN=ssid,JOBNAME=HMONssid,TRACE=ślad,REFRESH=nn
  ```

  Parametry TRACE i REFRESH są opcjonalne.

  nazwa_elementu

  Określa element biblioteki procedur wykonywany w celu uruchomienia monitora poprawności DB2, czyli programu DSNHMONP. Ta wpisana do katalogu procedura jest tworzona przez zadanie instalacji produktu DSNTIJHM.

  ssid

Określa nazwę lub identyfikator monitorowanego podsystemu DB2.

śledzenie

Określa flagę śledzenia. Możliwe są następujące wartości:

– ON - Włącz śledzenie. Rekordy śledzenia są zapisywane do standardowego wyjścia SYSOUT.

– OFF - Nie włączaj śledzenia.

Wartość domyślna to OFF.

nn

Określa godzinę (przy użyciu zegara 24-godzinnego), o której monitor poprawności odświeża obraz stanu harmonogramu oceniania używany do wyzwalania wartościowania strategii. Wartość domyślna to 22.

- Aby uruchomić wiele monitorów poprawności DB2, należy wprowadzić następującą komendę systemową START MVS:

  ```
  S nazwa_elementu
  ```

  nazwa_elementu

  Element biblioteki procedur wykonywany w celu uruchomienia wiele monitorów poprawności DB2, czyli programu DSNHMONA.

  Uwaga: Przed uruchomieniem wielu monitorów poprawności DB2 za pomocą jednej komendy START przy użyciu DSNHMONA zestaw danych HMONPARM określony w procedurze DSNHMONA musi zostać zapełniony listą monitorowanych podsystemów. Wpisana do katalogu procedura i zestaw danych są tworzone przez zadanie instalacji produktu DSNTIJHM.

- Aby odświeżyć obraz stanu harmonogramu wartościowania strategii używanego przez monitor poprawności produktu DB2 w celu określenia, kiedy wyzwolić wartościowanie strategii, należy wprowadzić komendę systemową MODIFY MVS:

  ```
  F HMONssid,APPL=REFRESH
  ```

  ssid

  Nazwa lub identyfikator podsystemu DB2 monitorowanego przez odświeżany monitor poprawności DB2.

- Aby zatrzymać monitor poprawności DB2, należy wydać następującą komendę STOP systemu MVS:

  ```
  STOP HMONssid lub P HMONssid
  ```

  ssid

  Nazwa lub identyfikator podsystemu DB2 monitorowanego przez zatrzymywany monitor poprawności DB2.

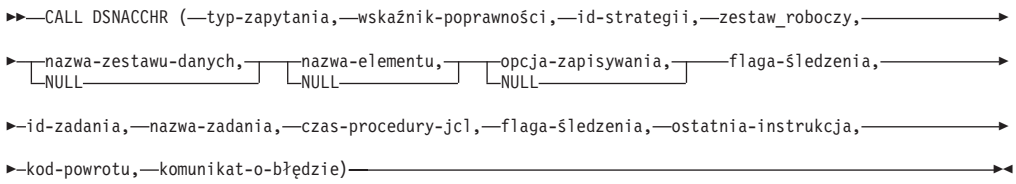
Wyświetlanie, wprowadzanie i zapisywanie zalecanych czynności

Aby wyświetlić, wprowadzić i zapisać działania zalecane dla obiektów alertu zidentyfikowanych w trakcie wartościowania strategii, należy wywołać procedurę składowaną DB2 SYSPROC.DSNACCHR, która została utworzona przez zadanie instalacyjne DSNTIJCC. Procedura DSNACCHR jest to procedura składowana określająca zalecone działania dla obiektów alertu zidentyfikowanych podczas wartościowania strategii i generująca zadanie JCL, które wykona te działania.

Poniższy diagram składni przedstawia instrukcję SQL CALL wywołującą procedurę DSNACCHR. Ponieważ konwencja łączenia dla procedury DSNACCHR to konwencja GENERAL WITH NULLS, jeśli parametry są przekazywane w zmiennych języka

macierzystego, dla każdej zmiennej języka macierzystego należy dopuścić wskaźnik null. Wskaźniki null dla wejściowych zmiennych języka macierzystego muszą zostać zainicjowane przed wykonaniem instrukcji CALL.

Składnia



typ-zapytania

Określa sposób postępowania z działaniami zalecanymi dla obiektów zidentyfikowanych jako będące w stanie alertu podczas wartościowania strategii. Możliwe są następujące wartości:

- 0 - Wyświetl zalecane działania dla obiektów alertu jako zadanie JCL
- 1 - Wprowadź zadanie JCL wykonujące zalecane działania dla obiektów alertu
- 2 - Wprowadź zadanie JCL wykonujące zalecane działania dla obiektów alertu i umieść zadanie w kolejce wstrzymania
- 3 - Zapisz zalecane działania dla obiektów alertu jako zadanie JCL w elemencie biblioteki

typ-zapytania jest to parametr wejściowy typu INTEGER.

wskaźnik-poprawności

Określa typ alertu uwzględniany przez procedurę DSNACCHR w zadaniu JCL. Możliwe są następujące wartości:

- RS - Stan ograniczony
- EX - Przekroczone przydziały
- RR - Wymagana instrukcja REORG
- CR - Wymagana instrukcja COPY
- RT - Wymagana instrukcja RUNSTATS
- SS - Wymagana instrukcja STOSPACE

wskaźnik-poprawności jest to parametr wejściowy typu VARCHAR(4).

id-strategii

Określa strategię konserwacji obiektów. *Id-strategii* jest to parametr wejściowy typu VARCHAR(7).

zestaw-roboczy

Określa zestaw roboczy strategii konserwacji obiektów identyfikujący obiekty alertu uwzględniane przez procedurę DSNACCHR w zadaniu JCL. Ten zestaw roboczy musi być identyfikowany za pomocą strategii i typu alertu określonych w parametrach *id-strategii* i *wskaźnik-poprawności*. *Zestaw-roboczy* jest to parametr wejściowy typu INTEGER.

nazwa-zestawu-danych

Określa pełną nazwę partycjonowanego zestawu danych (PDS, partitioned data set) lub rozszerzonego partycjonowanego zestawu danych (PDSE, partitioned data set extended). Ta wartość musi zostać określona, jeśli wartość parametru *typ-zapytania* jest równa 3. *Nazwa-zestawu-danych* jest to parametr wejściowy typu VARCHAR(44).

nazwa-elementu

Określa element rozszerzonego partycjonowanego zestawu danych (PDSE) określonego w parametrze *nazwa-zestawu-danych*, w którym zostanie zapisane zadanie JCL konserwacji obiektów. Ta wartość musi zostać określona, jeśli wartość parametru *typ-zapytania* jest równa 3. *Nazwa-elementu* jest to parametr wejściowy typu VARCHAR(8).

opcja-zapisu

Określa sposób zapisu zadania JCL konserwacji obiektów. Ta wartość musi zostać określona, jeśli wartość parametru *typ-zapytania* jest równa 3. Możliwe są następujące wartości:

- R - Zastąp
- A - Dołącz
- NM - Nowy element

Opcja-zapisu jest to parametr wejściowy typu VARCHAR(2).

flaga-śledzenia

Określa, czy śledzenie ma być włączone, czy wyłączone. Możliwe są następujące wartości:

- Y - Włącz śledzenie
- N - Wyłącz śledzenie

flaga-śledzenia jest to parametr wejściowy typu CHAR(1).

ID-zadania

Jeśli parametr *typ-zapytania* jest równy 1 lub 2, należy podać ID wprowadzanego zadania. *Id-zadania* jest to parametr wejściowy typu VARCHAR(8).

nazwa-zadania

Jeśli parametr *typ-zapytania* jest równy 1 lub 2, należy podać nazwę wprowadzanego zadania. *Nazwa-zadania* jest to parametr wyjściowy typu VARCHAR(8).

czas-procedury-jcl

Określa czas przetwarzania żądania. *czas-procedury-jcl* jest to parametr wyjściowy typu TIMESTAMP.

ostatnia-instrukcja

Jeśli procedura DSNACCHR zwraca poważny błąd (kod powrotu 12), to pole zawiera instrukcję SQL wykonywaną w chwili wystąpienia błędu. *Ostatnia-instrukcja* jest to parametr wyjściowy typu VARCHAR(2500).

kod-powrotu

Kod powrotu z procedury DSNACCHR. Możliwe są następujące wartości:

- 0 - Pomyślnie wykonano procedurę DSNACCHR

- 12 - Wykonanie procedury DSNACCHR zostało zakończone z poważnymi błędami. Parametr *komunikat-o-błędzie* zawiera komunikat opisujący błąd. Parametr *ostatnia-instrukcja* zawiera instrukcję SQL wykonywaną w chwili wystąpienia błędu.

Kod-powrotu jest to parametr wyjściowy typu INTEGER.

komunikat-o-błędzie

Jeśli procedura DSNACCHR zwraca poważny błąd (kod powrotu 12), to pole zawiera komunikaty o błędach włącznie ze sformatowanym obszarem SQLCA. *Komunikat-o-błędzie* jest to parametr wyjściowy typu VARCHAR(1331).

Procedura DSNACCHR zwraca jedną tabelę wynikową, gdy parametr *typ-zapytania* ma wartość 0. Tabela wynikowa zawiera zadanie JCL wygenerowane przez procedurę DSNACCHR. Tabela wynikowa procedury DSNACCHR jest tworzona przez zadanie instalacyjne DSNTIJCC. Tabela 12 przedstawia format tabeli wynikowej.

Tabela 12. Format tabeli wynikowej procedury DSNACCHR

Nazwa kolumny	Typ danych	Opis
JCLSEQNO	INTEGER	Numer kolejny wiersza tabeli (1,...,n)
JCLSTMT	VARCHAR(80)	Określa instrukcję JCL

Wyświetlanie podsumowań alertów poprawności

Funkcja HEALTH_OVERVIEW zwraca informacje z zestawu danych KSDS VSAM podsumowania alertu poprawności jako tabelę DB2. Ten zestaw danych jest tworzony przez zadanie instalacyjne DSNTIJHM.

Zestaw danych podsumowania alertu poprawności zawiera informacje o stanie monitora poprawności DB2 i statystyki podsumowania alertu dla każdego podsystemu DB2 uprzednio lub obecnie monitorowanego przez monitor poprawności danego systemu MVS lub klastra Parallel Sysplex. Te informacje są zwracane do klienta wraz z wierszem dla każdego podsystemu DB2 i zalecanym alertem.

Wynikiem działania funkcji jest tabela DB2 z następującymi kolumnami:

adres-ip

Adres IP serwera DB2. Jest to kolumna typu VARCHAR(40).

db2-ssid

Identyfikator podsystemu DB2. Jest to kolumna typu VARCHAR(4).

wskaźnik-poprawności

Typ alertu. Możliwe są następujące wartości:
- RS - Stan ograniczony
- EX - Przekroczone przydziały
- RR - Wymagana instrukcja REORG
- CR - Wymagana instrukcja COPY
- RT - Wymagana instrukcja RUNSTATS

- SS - Wymagana instrukcja STOSPACE
- PO - Nieudane wartościowanie strategii
- HM - Stan monitora poprawności

Wskaźnik-poprawności jest to kolumna typu VARCHAR(4).

nazwa-hosta

Pełna nazwa domeny serwera DB2. Jest to kolumna typu VARCHAR(255).

statystyki-podsumowania

Stan monitora poprawności DB2, jeśli parametr *wskaźnik-poprawności* jest równy 'HM'. Możliwe są następujące wartości:

- 0 Monitor poprawności nie został uruchomiony
- 1 Monitor poprawności został uruchomiony
- -1 Nieznany stan monitora poprawności

W przeciwnym razie łączna liczba obiektów alertu z typem alertu określonym w parametrze *wskaźnik-poprawności*. Jest to kolumna typu INTEGER.

stan-alertu

Stan alertu określonego w parametrze *wskaźnik-poprawności*. Możliwe są następujące wartości:

- 5 - Alarm
- 4 - Uwaga
- 3 - Ostrzeżenie
- 0 - Normalny

Stan-alertu jest zawsze równy 0, gdy parametr *wskaźnik-poprawności* ma wartość 'HM'. Jest to kolumna typu INTEGER.

Nazwa programu zewnętrznego dla funkcji jest równa HEALTH_OVERVIEW, a konkretna nazwa to DSNACC.DSNACCHO. Ta funkcja jest tworzona przez zadanie instalacyjne DSNTIJCC.

Przykład: Znajdź łączną liczbę obiektów alertu wymagających funkcji COPY dla podsystemu DB2 'ABCD':

```
SELECT SUMMARYSTATS FROM TABLE (DSNACC.HEALTH_OVERVIEW()) AS T
      WHERE DB2SSID = 'ABCD'
              AND HEALTHIND = 'CR';
```

Wyświetlanie obiektów alertów poprawności

Obiekty alertów zidentyfikowane podczas ostatniej pomyślnej iteracji wartościowania strategii są zapisywane w tych tabelach repozytorium obiektów alertów w zależności od typu obiektu.

Do obiektów alertów należą:

- DSNACC.HM_MAINT_TS dla obszarów tabel
- DSNACC.HM_MAINT_IX dla indeksów
- DSNACC.HM_MAINT_SG dla grup pamięci masowej

Program DB2 tworzy wiele widoków dla tych tabel repozytorium obiektów alertów. Widoki i tabele repozytorium obiektów alertów są tworzone przez zadanie instalacyjne DSNTIJCC. Tabela 13 przedstawia tabele, dla których są zdefiniowane widoki, i opisy widoków. Wszystkie nazwy widoków i nazwy tabel mają kwalifikator DSNACC.

Tabela 13. Widoki obiektów alertów poprawności

Nazwa widoku	Dla tabeli	Opis widoku
HM_ALERT_TS_RS	HM_MAINT_TS	Wyświetla wszystkie obszary tabel w stanie ograniczonym.
HM_ALERT_TS_EX	HM_MAINT_TS	Wyświetla wszystkie obszary tabel, których przydziały przekroczyły limit określony przez użytkownika.
HM_ALERT_TS_RR	HM_MAINT_TS	Wyświetla wszystkie obszary tabel wymagające wykonania instrukcji REORG.
HM_ALERT_TS_CR	HM_MAINT_TS	Wyświetla wszystkie obszary tabel wymagające wykonania instrukcji COPY.
HM_ALERT_TS_RT	HM_MAINT_TS	Wyświetla wszystkie obszary tabel wymagające wykonania instrukcji RUNSTATS.
HM_ALERT_IX_RS	HM_MAINT_IX	Wyświetla wszystkie indeksy w stanie ograniczonym.
HM_ALERT_IX_EX	HM_MAINT_IX	Wyświetla wszystkie indeksy, których przydziały przekroczyły limit określony przez użytkownika.
HM_ALERT_IX_RR	HM_MAINT_IX	Wyświetla wszystkie obszary indeksów wymagające wykonania instrukcji REORG.
HM_ALERT_IX_CR	HM_MAINT_IX	Wyświetla wszystkie indeksy wymagające wykonania instrukcji COPY.
HM_ALERT_IX_RT	HM_MAINT_IX	Wyświetla wszystkie indeksy wymagające wykonania instrukcji RUNSTATS.
HM_ALERT_SG_SS	HM_MAINT_SG	Wyświetla wszystkie grupy pamięci masowej wymagające wykonania instrukcji STOSPACE.

Część 3. Wysoka dostępność i program DB2 Connect

Przedstawiono tu rozważania dotyczące korzystania z opcji wysokiej dostępności w środowiskach korzystających z produktu DB2 Connect. Jeśli znajdujący się w sieci serwer bazy danych stanie się niedostępny, ważne jest posiadanie możliwości przekierowania klienckiej stacji roboczej do alternatywnego serwera bazy danych w sieci.

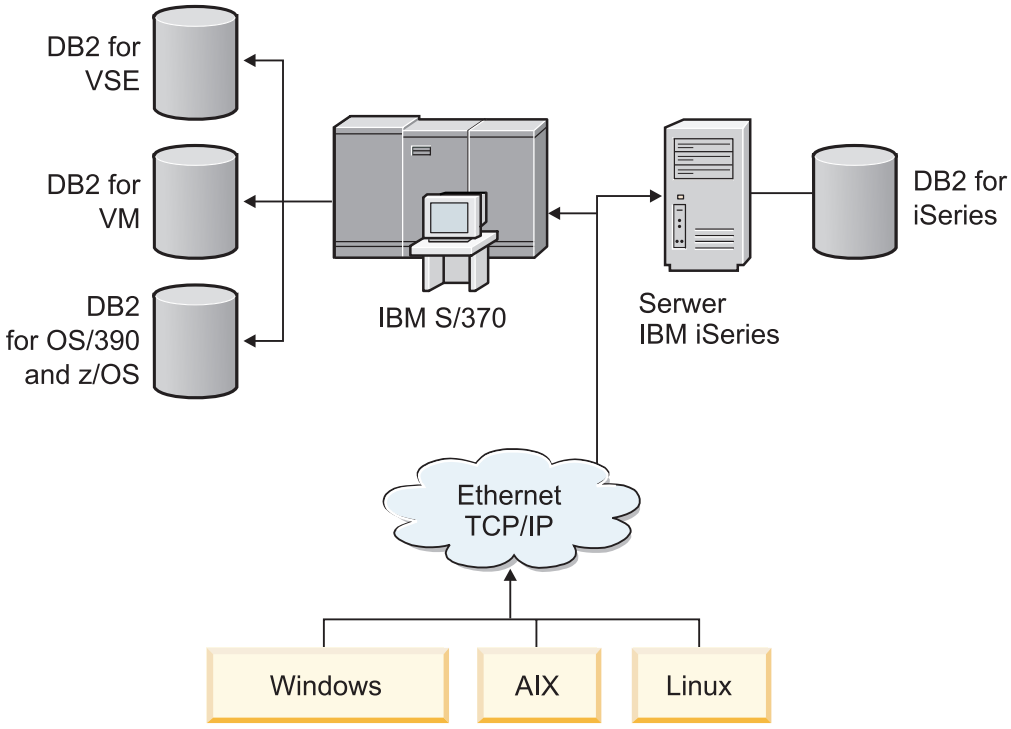
© Copyright IBM Corp. 1993, 2010

Rozdział 11. Wysoka dostępność i równoważenie obciążenia w połączeniach z bazą danych na hoście

Na współczesnym rynku technologii informacyjnych istnieje duże zapotrzebowanie na rozwiązania gwarantujące nieprzerwaną dostępność danych. Zapotrzebowanie to musi zostać spełnione, aby firmy mogły zachować konkurencyjność i stale się rozwijać. Wiele współczesnych aplikacji WWW, e-biznesowych czy arkuszy kalkulacyjnych wymaga dostępu do danych przedsiębiorstwa. W tym celu konieczne jest nawiązywanie niezawodnych, szybkich i bezpiecznych połączeń z bazami danych na hostach lub serwerach System i. Takie połączenie musi być stale dostępne i zdolne do obsługi wysokiego zapotrzebowania na połączenia w warunkach krytycznego obciążenia. Jak zatem budować takie połączenia?

Scenariusz wysokiej dostępności

Firma dysponuje szeregiem stacji roboczych i serwerów aplikacji działających pod kontrolą systemów operacyjnych Windows, Linux i UNIX. Komputery te wymagają dostępu do danych znajdujących się na kilku hostach i w kilku bazach danych System i. Aplikacje działające na tych komputerach potrzebują szybkich i niezawodnych połączeń z bazami danych. Cały system jest połączony siecią Ethernet wykorzystującą protokół TCP/IP.



Rysunek 11. Przykładowy scenariusz z użyciem sieci.

Aby stacje robocze i serwery aplikacji miały dostęp do baz danych na hostach i serwerach System i, konieczne jest pośrednictwo elementu obsługującego połączenia. Komponent ten musi zapewniać połączenia z bazami danych na hostach lub serwerach System i charakteryzujące się wysoką dostępnością, stabilnością i szybkością. Musi on także być skalowalny w zakresie przewidywanego przyszłego wzrostu liczby połączeń.

© Copyright IBM Corp. 1993, 2010

Aby uzyskać szczegółowe informacje dotyczące rozwiązania korzystającego z produktu DB2 Connect i opcji automatycznego przekierowywania klientów, użyj odsyłaczy do stron pokrewnych.

Rozdział 12. Opis i konfiguracja automatycznego przekierowywania klientów (DB2 Connect)

Głównym celem opcji automatycznego przekierowywania klientów jest umożliwienie aplikacji IBM Data Server Client odzyskania połączenia po jego utracie, dzięki czemu aplikacja może kontynuować swoją pracę z możliwie najkrótszymi przerwami. Jak wynika to z nazwy, przekierowywanie ma podstawowe znaczenie dla obsługi operacji ciągłych. Ale przekierowanie jest możliwe tylko w sytuacji, gdy istnieje alternatywne miejsce znane w połączeniu klienta. W środowisku wysokiej dostępności, w którym nie ma serwera DB2 Connect, synchronizacja bazy danych, do której jest uzyskiwany dostęp, realizowana jest zwykle pomiędzy oryginalnym serwerem DB2 a alternatywnym serwerem DB2 przy użyciu jednej z dostępnych metod, takich jak HACMP (High availability cluster multiprocessor) lub HADR (High availability disaster recovery).

Jednak ze względu na brak wymogu synchronizacji lokalnych baz danych w przypadku serwera DB2 Connect trzeba jedynie zapewnić, aby zarówno oryginalny, jak i alternatywny serwer DB2 Connect miały docelową bazę danych na docelowym hoście lub w systemie System i wpisaną do katalogu w taki sposób, żeby była dostępna za pośrednictwem identycznych aliasów bazy danych.

Uwaga: W środowisku serwera DB2 Connect można określić alternatywny serwer DB2 Connect, aby włączyć możliwość automatycznego przekierowywania pomiędzy klientem a serwerem DB2 Connect. Aby wystąpiło przekierowywanie pomiędzy produktem DB2 Connect a serwerem bazy danych hosta lub systemu System i, serwer zdalny musi udostępniać co najmniej jeden adres alternatywny. W przypadku programu DB2 for z/OS znanych jest wiele adresów jeśli baza danych jest środowiskiem współużytkowania danych Sysplex.

Jeśli obsługa środowiska Sysplex jest włączona, możliwość przekierowywania w środowisku Sysplex może być skonfigurowana pomiędzy programem DB2 Connect a serwerem bazy danych hosta. Możliwość przekierowywania w środowisku Sysplex jest funkcją programu DB2 Connect, umożliwiającą programowi DB2 Connect ponawianie połączeń z innymi elementami środowiska Sysplex w przypadku utraty połączenia z oryginalnym elementem. Alternatywny serwer nie musi być wpisany do katalogu bazy danych, aby możliwe włączenie możliwości przekierowywania w środowisku Sysplex w programie DB2 Connect. Domyślnie możliwość przekierowywania w środowisku Sysplex jest włączona, jeśli włączona jest obsługa środowiska Sysplex.

Aby program IBM Data Server Client miał zdolność odtwarzania po przerwaniu komunikacji z serwerem DB2 Connect przy użyciu automatycznego przekierowywania klienta, położenie alternatywnego serwera DB2 Connect musi zostać określone, zanim wystąpi przerwa w komunikacji. Do zdefiniowania położenia alternatywnego serwera DB2 Connect dla określonej bazy danych na systemie hostowym lub bazy danych w systemie System i służy komenda UPDATE ALTERNATE SERVER FOR DATABASE. Alternatywna nazwa hosta i numer portu stanowią część komendy. Położenie jest zapamiętywane w pliku systemowego katalogu bazy danych na serwerze DB2 Connect. Aby zapewnić, że określone położenie alternatywnego serwera DB2 Connect dotyczy tej bazy danych dla wszystkich klientów, położenie alternatywnego serwera musi zostać określone po stronie serwera DB2 Connect. Jeśli jest określone w instancji klienta, alternatywny serwer jest ignorowany.

Na przykład załóżmy, że baza danych na hoście lub w systemie System i jest wpisana do katalogu przy użyciu aliasu bazy danych db1 na serwerze DB2 Connect S1 (o nazwie hosta

© Copyright IBM Corp. 1993, 2010

db2conn1 i numerze portu 122). Administrator bazy danych chce określić alternatywny serwer DB2 Connect S2 o nazwie hosta db2conn2 i numerze portu 123. W takiej sytuacji administrator bazy danych powinien wykonać na serwerze DB2 Connect S1 następującą komendę:

```
db2 update alternate server for database db1 using hostname db2conn2 port 123
```

Po określeniu położenia alternatywnego serwera DB2 Connect dla aliasu db1 bazy danych na serwerze DB2 Connect S1 informacje te są zwracane do klienta IBM Data Server Client w trakcie procesu nawiązywania połączenia. Jeśli komunikacja między klientem IBM Data Server Client a serwerem DB2 Connect S1 zostanie z jakiegokolwiek powodu przerwana (zwykle na skutek błędu komunikacyjnego, takiego jak błąd SQL o kodzie 30081 lub o kodzie 1224), klient IBM Data Server Client będzie usiłował ponownie nawiązać połączenie z db1 za pośrednictwem oryginalnego serwera DB2 Connect (S1) albo alternatywnego serwera DB2 Connect (S2). Próby nawiązania połączenia będą podejmowane na przemian w stosunku do jednego lub drugiego serwera. Kolejne próby podejmowane są natychmiast, a odstęp czasu między nimi stopniowo się wydłuża.

Po pomyślnym nawiązaniu połączenia zwracany jest kod SQL -30108 wskazujący, że połączenie z bazą danych zostało nawiązane ponownie po wystąpieniu błędu komunikacji. Zwracana jest nazwa hosta lub adres IP i nazwa usługi lub numer portu. Klient IBM Data Server Client zwraca błąd w przypadku awarii oryginalnego połączenia tylko wtedy, gdy próba ponownego nawiązania połączenia ze strony klienta jest niemożliwa zarówno w odniesieniu do oryginalnego, jak i alternatywnego serwera.

Należy także zwrócić uwagę na następujące kwestie dotyczące połączeń z serwerem alternatywnym w środowisku serwera DB2 Connect:

- Kiedy serwer DB2 Connect jest wykorzystywany do udostępniania bazy danych na systemie hostowym lub w systemie System i na rzecz zarówno zdalnych, jak i lokalnych klientów, może się pojawić niejasność dotycząca informacji o połączeniu z serwerem alternatywnym zapisanych w pozycji katalogu systemowej bazy danych. Aby zminimalizować to nieporozumienie, należy rozważyć zapisanie dwóch wpisów w systemowym katalogu bazy danych reprezentujących tę samą bazę danych hosta lub serwera System i. Jeden wpis będzie dotyczyć klientów zdalnych, a drugi klientów lokalnych.
- Wszystkie informacje na temat SYSPLEX, które są zwracane z docelowego serwera DB2 for z/OS, są przechowywane wyłącznie w pamięci podręcznej serwera DB2 Connect. Tylko informacje z serwera alternatywnego zapisywane są na dysk. Jeśli istnieje wiele serwerów alternatywnych lub wiele serwerów aktywnych, informacje przechowywane są tylko w pamięci i po zakończeniu procesu następuje ich utrata.

Rozdział 13. Konfigurowanie automatycznego przekierowywania klientów dla technologii dystrybucji połączeń klientów

Technologie dystrybucji i rozsyłania, takie jak WebSphere EdgeServer, dystrybuują żądania ponownego połączenia aplikacji klienckich do zdefiniowanego zestawu systemów w sytuacji, gdy zawiedzie podstawowy serwer bazy danych. W przypadku korzystania z technologii dystrybucji razem z funkcją automatycznego przekierowywania klientów DB2 dystrybutor należy zidentyfikować jako serwer alternatywny dla automatycznego przekierowywania klientów DB2.

Technologii dystrybucji można używać w środowisku zbliżonym do poniższego:

Klient —> technologia dystrybucji —> (serwer DB2 Connect Server 1 lub serwer DB2 Connect Server 2) —> DB2 z/OS

gdzie:

- Komponent technologii dystrybucji ma nazwę hosta TCP/IP DTnazwa_hosta.
- Serwer DB2 Connect Server 1 ma nazwę hosta TCP/IP GWYnazwa_hosta1.
- Serwer DB2 Connect Server 2 ma nazwę hosta TCP/IP równą GWYnazwa_hosta2.
- Nazwa hosta TCP/IP serwera DB2 z/OS to zOSnazwa_hosta.

Klient jest wpisywany do katalogu z nazwą **DTnazwa_hosta**, aby można było wykorzystać technologię dystrybucji w celu uzyskania dostępu do dowolnego z serwerów DB2 Connect. Technologia dystrybucji podejmuje decyzję o użyciu serwera **GWYnazwa_hosta1** lub **GWYnazwa_hosta2**. Po podjęciu decyzji klient uzyskuje bezpośrednie połączenie przez gniazdo z jednym z dwóch gatewayów DB2 Connect. Po nawiązaniu połączenia przez gniazdo z wybranym serwerem DB2 Connect powstaje typowe połączenie klienta z serwerem DB2 Connect oraz systemem DB2 z/OS.

Dla przykładu można przyjąć, że program dystrybucyjny wybrał serwer **GWYnazwa_hosta2**. Tworzy to następujące środowisko:

Klient —> Serwer 2 DB2 Connect —> DB2 z/OS

Program dystrybucyjny nie wznawia żadnego połączenia, jeśli wystąpi jakakolwiek awaria komunikacji. Jeśli dla bazy danych w takim środowisku należy włączyć opcję automatycznego przekierowywania klientów, alternatywny serwer dla powiązanej bazy danych lub baz danych na serwerze DB2 Connect (serwer DB2 Connect Server 1 lub serwer DB2 Connect Server 2) powinien zostać ustawiony jako program dystrybucyjny (DTnazwa_hosta). Następnie, jeśli serwer DB2 Connect Server 1 zablokuje się z jakiegokolwiek powodu, zostanie wyzwolone automatyczne przekierowywanie klientów i nastąpi próba połączenia klienta z programem dystrybucyjnym zarówno dla serwera podstawowego, jak i alternatywnego. Ta opcja umożliwia zachowanie możliwości programu dystrybucyjnego i połączenie jej z opcją automatycznego przekierowywania klientów DB2. Ustawienie serwera alternatywnego na nazwę hosta inną niż nazwa hosta programu dystrybucyjnego nadal pozostawia klientom możliwość korzystania z opcji automatycznego przekierowywania klientów Jednak klienty będą nawiązywać bezpośrednie połączenia ze zdefiniowanym serwerem alternatywnym, pomijając tym samym technologię dystrybucji, co spowoduje wyeliminowanie programu dystrybucyjnego i zapewnianych przez niego korzyści.

© Copyright IBM Corp. 1993, 2010

Opcja automatycznego przekierowywania klientów powoduje przechwytywanie następujących kodów SQL:

- sqlcode -20157
- sqlcode -1768 (kod przyczyny = 7)

Uwaga: Opcja przekierowywania klientów może nie być w porę informowana o awariach gniazd, jeśli parametr konfiguracyjny systemu operacyjnego "TCP Keepalive" ma zbyt dużą wartość (nazwa tego parametru konfiguracyjnego jest różna w zależności od platformy).

Część 4. Strojenie i program DB2 Connect

W środowisku bazy danych korzystającym z programu DB2 Connect do przenoszenia żądań i odpowiedzi baz danych pomiędzy klienckimi stacjami roboczymi a serwerami baz danych istnieją pewne uwarunkowania dotyczące wydajności. Istnieje kilka metod zwiększenia lub zachowania wydajności w takim środowisku.

© Copyright IBM Corp. 1993, 2010

Rozdział 14. Uwagi dotyczące wydajności programu DB2 Connect

Wydajność to sposób działania systemu komputerowego przy określonym obciążeniu. Jest ona zależna od dostępnych zasobów oraz sposobu ich wykorzystania i współużytkowania. Aby poprawić wydajność, najpierw należy ją określić.

Można wybrać wiele różnych *charakterystyk wydajności*, na przykład:

Czas odpowiedzi
Czas między momentem wysłania przez aplikację żądania do bazy danych a momentem otrzymania odpowiedzi.

Przepustowość transakcyjna
Liczba jednostek pracy, które można wykonać w jednostce czasu. Jednostka pracy może być prosta, na przykład pobieranie i aktualizowanie wiersza, lub złożona wywołująca setki instrukcji języka SQL.

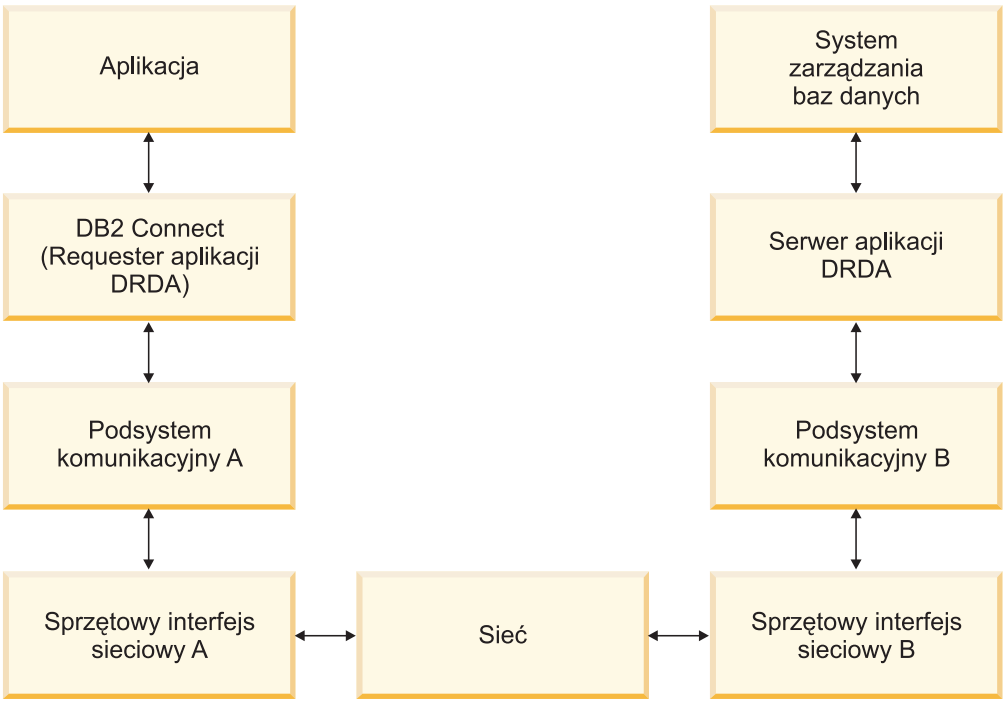
Szybkość przesyłania danych
Liczba bajtów danych przesłanych między aplikacją DB2 Connect a bazą danych na hoście lub serwerze System i w jednostce czasu.

Wydajność jest ograniczona przez dostępny sprzęt i oprogramowanie. Procesor, pamięć i adaptery sieciowe są przykładami zasobów sprzętowych. Przykładami zasobów programowych są podsystemy komunikacyjne, podsystemy stronicowania, moduł mbuf dla systemu AIX.

Przepływy danych

Rys. 12 na stronie 96 przedstawia ścieżkę przepływu danych między serwerem bazy danych na hoście lub serwerze System i a stacją roboczą za pośrednictwem programu DB2 Connect.

© Copyright IBM Corp. 1993, 2010



Rysunek 12. Przepływ danych w programie DB2 Connect.

- Baza danych na hoście lub serwerze System i oraz część podsystemu komunikacyjnego B działają zwykle w tym samym systemie. System składa się z jednej lub większej liczby jednostek centralnych, pamięci głównej, podsystemu we/wy, DASD oraz systemu operacyjnego. Ponieważ inne programy mogą współużytkować te komponenty, rywalizacja o zasoby może powodować problemy związane z wydajnością.
- Sieć składa się z kabli, węzłów, linii komunikacyjnych, przełączników i innych sterowników komunikacji. Sieciowym interfejsem sprzętowym B mogą być na przykład kontrolery komunikacyjne 3745 lub 3172 lub adapter Token Ring dla serwera System i. Sprzętowe interfejsy A i B mogą być połączone przy użyciu kilku nośników transmisji.
- Sprzętowym interfejsem sieciowym A mogą być: adaptery Token Ring, Ethernet**, inne adaptery LAN lub adaptery, które obsługują sterowanie synchronicznym łączem danych (SDLC) lub protokoły X.25.
- Program DB2 Connect i podsystem komunikacyjny A znajdują się zwykle w tym samym systemie. Na potrzeby niniejszego omówienia założono, że aplikacja znajduje się także w tym samym systemie.

Wąskie gardła

Przepustowość transakcji zależy od najwolniejszego komponentu w systemie. Jeśli takie wąskie gardło zostanie zidentyfikowane, często można zmniejszyć istotność problemu przez zmianę parametrów konfiguracyjnych, przydzielenie większej ilości zasobów do komponentu powodującego występowanie problemu, zaktualizowanie komponentu lub dodanie nowego komponentu, który przejmie (rozładuje) część pracy.

Do określenia czasu, jakiego potrzebuje zapytanie dla każdego komponentu, można zastosować rozmaite narzędzia. Pomoże to w podjęciu decyzji, które komponenty należy dostroić lub zaktualizować do nowszej wersji w celu poprawienia wydajności. Jeśli na przykład okaże się, że zapytanie poświęca 60% czasu dla komputera z programem DB2 Connect, można dostroić DB2 Connect lub (w przypadku istnienia klientów zdalnych) dodać inny komputer z programem DB2 Connect do sieci.

Testowanie porównawcze

Testowanie porównawcze pozwala porównać wydajność w jednym środowisku z wydajnością w innym środowisku. Testowanie porównawcze można rozpocząć, uruchamiając aplikację testową w normalnym środowisku. Ponieważ problemy wydajności dotyczą tylko pewnej wąskiej grupy parametrów, należy ograniczyć zakres funkcji, które będą testowane i obserwowane.

Testowanie porównawcze nie musi być skomplikowane. Aby otrzymać wartościowe informacje, nie jest wymagane emulowanie całej aplikacji. Należy uruchomić proste pomiary i zwiększyć ich złożoność tylko w uzasadnionych przypadkach.

Dobre testowanie porównawcze wyróżnia się następującymi cechami:

- Każdy test jest powtarzalny.
- Każda iteracja testu jest uruchamiana w tym samym stanie systemu.
- Sprzęt i oprogramowanie używane do testowania porównawczego są zgodne ze środowiskiem produkcyjnym.
- Poza testowanym oprogramowaniem w systemie nie są aktywne żadne inne funkcje i aplikacje, chyba że scenariusz testu uwzględnia inne działania w systemie.

  Uwaga: Uruchomione aplikacje zajmują pamięć, nawet jeśli ich działanie zostało zminimalizowane lub są one bezczynne. Może to powodować operacje stronicowania i pogorszenie wyników testowania porównawczego.

Narzędzia do pomiaru wydajności

W następujących tabelach przedstawiono niektóre narzędzia pomocne przy pomiarze wydajności systemu. Ponieważ same narzędzia wykorzystują zasoby systemu, nie mogą być aktywne przez cały czas.

Tabela 14. Narzędzia do pomiaru wydajności procesora i pamięci

System	Narzędzie	Opis
AIX	vmstat, time, ps, tprof	Dostarcza informacji dotyczących problemów związanych z jednostką centralną oraz zawartością pamięci na stacji roboczej i klientach zdalnych DB2 Connect.
HP-UX	vmstat, time, ps, monitor i glance (jeśli jest dostępny)	
Windows	Monitor wydajności firmy Microsoft	

Tabela 15. Narzędzia do pomiaru wydajności bazy danych

System	Narzędzie	Opis
Wszystkie	Monitor bazy danych	Określa, czy źródłem problemu jest baza danych.

Tabela 15. Narzędzia do pomiaru wydajności bazy danych (kontynuacja)

System	Narzędzie	Opis
OS/390 lub zSeries	IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS, ASG-TMON for DB2 (ASG) oraz CA Insight Performance Monitor for DB2 for z/OS (Computer Associates International, Inc.)	
Windows	Monitor wydajności firmy Microsoft	

Tabela 16. Narzędzia do pomiaru wydajności sieci

System	Narzędzie	Opis
AIX	netpmon	Wykonuje raporty statystyczne dla sieci niskiego poziomu, włączając w to statystyki TCP/IP, na przykład liczbę pakietów lub ramek na sekundę.
Sterownik sieciowy (np. 3745)	Monitor wydajności NetView	Wykonuje raporty wykorzystania komunikacyjnych elementów sterujących i metody VTAM.
Linux i UNIX	netstat	Obsługuje ruch TCP/IP.

Rozdział 15. Optymalizacja dostępu ODBC

Baza danych DB2 oferuje specjalną optymalizację zaprojektowaną w taki sposób, aby poprawić wydajność komunikacji za pośrednictwem interfejsu ODBC. Te rozszerzenia są dostępne dla programów Microsoft Access, Lotus Approach i Visual Basic. Za pomocą Asysty podczas konfigurowania (CA, Configuration Assistant) w programie DB2 można poprawić przepustowość sterowników ODBC.

Aby aktywować zoptymalizowaną łączność ODBC:

- Jeśli definiujesz nowe połączenie:
  1. Uruchom moduł CA produktu DB2.
  2. Otwórz menu Wybrane i wybierz opcję Dodaj bazę danych za pomocą kreatora.
  3. Przechodź przez kolejne strony kreatora, aż do strony **Źródło danych**.
  4. Zaznacz opcję **Zarejestruj bazę danych dla CLI/ODBC**.
  5. Określ sposób, w jaki aplikacje CLI/ODBC uzyskujące dostęp do tej bazy danych powinny być rejestrowane:
     - **Jako systemowe źródło danych** oznacza, że baza danych jest dostępna dla wszystkich użytkowników w systemie.
     - **Jako źródło danych użytkownika** oznacza, że jesteś jedynym użytkownikiem, który może uzyskać dostęp do bazy danych.
     - **Jako plikowe źródło danych** oznacza, że zostanie utworzony plik zawierający informację o źródle danych. To plikowe źródło danych może być współużytkowane z innymi stacjami roboczymi, jeśli dostępne jest połączenie TCP/IP. W przeciwnym razie plik może być używany tylko na tym komputerze.
  6. Wpisz wartość w polu **Nazwa źródła danych**.
  7. (Opcjonalnie) Wybierz aplikację z listy **Optymalizuj dla aplikacji**, aby zoptymalizować ustawienia źródła danych dla konkretnej aplikacji.
  8. Kliknij przycisk **OK** i zakończ pracę z Asystą podczas konfigurowania.
- Jeśli aktualizujesz istniejące połączenie:
  1. Uruchom moduł CA produktu DB2.
  2. Kliknij dwukrotnie alias bazy danych, która ma zostać zoptymalizowana.
  3. Kliknij przycisk **Źródło danych**.
  4. Zaznacz opcję **Zarejestruj bazę danych dla CLI/ODBC**.
  5. Określ sposób, w jaki aplikacje CLI/ODBC uzyskujące dostęp do tej bazy danych powinny być rejestrowane:
     - **Jako systemowe źródło danych** oznacza, że baza danych jest dostępna dla wszystkich użytkowników w systemie.
     - **Jako źródło danych użytkownika** oznacza, że jesteś jedynym użytkownikiem, który może uzyskać dostęp do bazy danych.
     - **Jako plikowe źródło danych** oznacza, że zostanie utworzony plik zawierający informację o źródle danych. To plikowe źródło danych może być współużytkowane z innymi stacjami roboczymi, jeśli dostępne jest połączenie TCP/IP. W przeciwnym razie plik może być używany tylko na tym komputerze.
  6. Wpisz wartość w polu **Nazwa źródła danych**.
  7. (Opcjonalnie) Wybierz aplikację z listy **Optymalizuj dla aplikacji**, aby zoptymalizować ustawienia źródła danych dla konkretnej aplikacji.
  8. Kliknij przycisk **OK** i zakończ pracę z Asystą podczas konfigurowania.

© Copyright IBM Corp. 1993, 2010

Rozdział 16. Projektowanie aplikacji

Podczas tworzenia aplikacji można poprawić jej wydajność na kilka sposobów.

Złożone instrukcje SQL i procedury składowane

Dla aplikacji, które wysyłają i otrzymują wiele komend i odpowiedzi, istotne znaczenie ma narzut działań w sieci. Można go zredukować przy użyciu złożonych instrukcji SQL i procedur składowanych.

Jeśli aplikacja wysyła wiele instrukcji języka SQL, które nie są ze sobą logicznie powiązane, można używać złożonych instrukcji SQL. Jeśli wewnątrz grupy mają być użyte instrukcje powiązane logicznie, można skorzystać z procedur składowanych.

Wewnątrz złożonych instrukcji SQL można używać wszystkich instrukcji wykonywalnych poza następującymi:

```
CALL
FETCH
CLOSE
OPEN
Compound SQL
Connect
Prepare
hosta
Describe
Rollback
Disconnect
Set connection
execute immediate
```

Procedury składowane ograniczają przepływ danych w sieci przez umieszczenie oprogramowania na serwerze. Możliwe jest zatwierdzanie automatyczne przy zakończeniu procedury. Można również zwracać tabele wynikowe minimalizujące logikę aplikacji po stronie klienta.

Grupowanie żądań

Grupowanie żądań związanych z bazą danych (instrukcje języka SQL) w jedno żądanie może zredukować liczbę żądań i odpowiedzi przesyłanych przez sieć.

Na przykład zgrupowanie następujących instrukcji:

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=2
```

w

```
SELECT COL1, COL2, COL5, COL6 FROM TABLEA WHERE ROW_ID=1 OR ROW_ID=2
```

spowoduje wysłanie mniejszej liczby żądań przez sieć.

Aby zredukować liczbę zwracanych wierszy, można także używać parametrów, takich jak IN i BETWEEN. Ponadto w instrukcjach UPDATE i DELETE można używać parametrów WHERE, IN i BETWEEN.

Logika predykatów

Logiki predykatów można używać w wypadku żądań dotyczących wyłącznie potrzebnych wierszy i kolumn. Zminimalizuje to ruch w sieci oraz narzut dla jednostki centralnej związany z przesyłaniem danych.

Nie należy na przykład używać zapytania:

© Copyright IBM Corp. 1993, 2010

```
SELECT * FROM TABLEA
```

jeśli jest potrzebny tylko pierwszy wiersz tabeli TABLEA z ROW_ID=1 albo tylko kolumny 1 i 2.

Łączenie danych w bloki

Łączenia danych w bloki należy używać w przypadku większej ilości danych z serwera. Łączenie w bloki poprawia wykorzystanie pasma sieci i ogranicza nakład pracy procesora zarówno serwera bazy danych na hoście lub serwerze System i, jak i serwera DB2 Connect. Bez względu na wielkość komunikatu, ilość informacji sterujących CPU i sieci dla każdego wysłanego i otrzymanego komunikatu jest stała. Łączenie danych w bloki redukuje liczbę komunikatów wymaganych dla tej samej wielkości przesyłanych danych.

Jeśli łączenie w bloki jest wykonywane, pierwszy wiersz danych z zapytania nie będzie dostarczony do aplikacji, dopóki pierwszy blok nie zostanie odebrany. Łączenie w bloki zwiększa czas wyszukiwania pierwszego wiersza, ale zmniejsza czas wyszukiwania kolejnych wierszy.

Inne rozważania dotyczą wielkości używanej pamięci. Wykorzystywana pamięć wzrasta, jeśli zostaje włączona funkcja łączenia w bloki.

W programie DB2 Connect można regulować ilość danych przesyłanych w każdym bloku.

Aby wywołać łączenie w bloki, należy użyć opcji BLOCKING w komendzie prep lub bind. Łączenie w bloki jest włączone, jeśli:

- kursor jest tylko do odczytu lub
- kursor jest niejednoznaczny i łączenie w bloki zostaje określone podczas wykonywania komendy prep lub bind.

Uwaga: Kursor jest zawsze niejednoznaczny, jeśli używany jest dynamiczny SQL.

Instrukcje SQL z parametrem BLOCKING

Aktualizowalne instrukcje SELECT (używające instrukcji UPDATE/DELETE WHERE CURRENT OF) są zapytaniami niełączącymi danych w bloki, więc można ich używać tylko wtedy, gdy jest to bezwzględnie konieczne.

Aktualizowalna instrukcja SELECT zapewnia, że między momentem zakończenia instrukcji SELECT i wywołania instrukcji UPDATE/DELETE wiersz nie zostanie zmieniony. Jeśli dla danej aplikacji nie jest ważny poziom współbieżności, alternatywnym rozwiązaniem jest zastosowanie instrukcji DELETE lub UPDATE z kryterium wyszukiwania opartym na wartościach zwracanych z nieaktualizowalnej instrukcji SELECT.

W wypadku instrukcji SELECT w trybie tylko do odczytu należy określić klauzulę FOR FETCH ONLY; nie dotyczy to systemów VM i VSE, w których klauzula ta nie jest obsługiwana.

Statyczny i dynamiczny SQL

Statycznego SQL należy używać tak często, jak to możliwe. Uniknie się w ten sposób przygotowania sekcji SQL czasu wykonywania i kursorów niejednoznacznych. Jeśli nie można uniknąć stosowania dynamicznego SQL, w celu zminimalizowania ruchu w sieci i poprawy wydajności można wykonać następujące czynności:

- Przygotowując instrukcję SELECT, należy wykonać instrukcję PREPARE ... INTO obszar danych SQL. Należy przydzielić obszar danych SQL wystarczająco duży, aby mógł pomieścić wszystkie wybrane dane. Jeśli maksymalna liczba

kolumn jest równa *x* i nie planuje się jej zmiany, należy przydzielić obszar danych SQL dla *x* wartości SQLVAR. Jeśli nie można oszacować liczby kolumn (i wielkość pamięci nie stanowi problemu), należy użyć maksymalnej liczby SQLVAR (256).

Jeśli przydzielony obszar danych SQL nie jest dostatecznie duży do przechowywania zwracanego obszaru danych SQL, program musi ponownie wywołać instrukcję DESCRIBE z dostatecznie dużym obszarem danych SQL do ponownego przechowywania wyniku. Zwiększy to ruch w sieci.

Nie należy używać sekwencji PREPARE i DESCRIBE. Zastosowanie w ich miejsce instrukcji PREPARE.....INTO zapewni lepszą wydajność.

- Zamiast dynamicznych instrukcji COMMIT lub ROLLBACK należy wykonywać statycznie powiązane instrukcje SQL COMMIT lub ROLLBACK.
- Dla instrukcji innych niż SELECT, COMMIT lub ROLLBACK należy w celu wykonania instrukcji wywołać EXECUTE IMMEDIATE, a nie sekwencję PREPARE i EXECUTE.
- Aplikacje ODBC używają dynamicznego języka SQL. W celu poprawienia wydajności można użyć opcji statycznego profilowania CLI/ODBC. Opcja ta umożliwia przechwytywanie i przekształcanie wywołań ODBC na statyczne instrukcje, zapisywane w pakiecie bazy danych. Wydajność uzyskana w danym momencie zależy od złożoności aplikacji.

Inne uwagi dotyczące języka SQL

Zastosowanie procesora CLP (Procesor wiersza komend) powoduje z reguły, że program działa wolniej niż w przypadku dynamicznych instrukcji SQL, ponieważ procesor CLP musi przeprowadzić analizę danych wejściowych przed przekazaniem instrukcji SQL do silnika bazy danych. Ponadto procesor CLP formatuje odbierane dane, co nie jest konieczne w przypadku każdej aplikacji.

Instrukcje SQL w języku interpretowanym, takim jak REXX, działają wolniej niż te same instrukcje SQL w języku kompilowanym, takim jak C.

Istnieją dwa typy instrukcji CONNECT, typ 1 i typ 2. Przy użyciu połączenia typu 2, połączenie z bazą danych wprowadza poprzednie połączenie w stan uśpienia, lecz go nie usuwa. Jeśli przełączenie do uśpionego połączenia nastąpi później, uniknie się ładowania bibliotek i ustawiania struktur danych wewnętrznych. Z tego powodu użycie połączenia typu 2 może poprawić wydajność aplikacji, które mają dostęp do więcej niż jednej bazy danych.

Rozdział 17. Zarządzanie połączeniami

Na zarządzanie połączeniami składają się dwa elementy: pule połączeń i koncentrator połączeń. Pule połączeń pomagają ograniczyć nakład pracy związany z obsługą połączeń z bazą danych i zarządzać znaczną ilością połączeń. Koncentrator połączeń zwiększa skalowalność środowiska roboczego poprzez optymalizację zasobów używanych przez serwery baz danych hosta.

Zestawianie połączeń

Produkty serwerowe DB2 Connect, takie jak DB2 Connect Enterprise Edition, często zapewniają połączenia z bazą danych dla tysięcy jednoczesnych żądań klientów. Nawiązywanie i zrywanie połączeń z serwerem bazy danych może być procesem wiążącym się z dużym zapotrzebowaniem na zasoby, który niekorzystnie wpływa na wydajność zarówno serwera bazy danych, jak i serwera DB2 Connect.

Problem ten jest szczególnie widoczny w środowiskach sieci WWW, gdzie każda wizyta na stronie WWW może wymagać nawiązania nowego połączenia z serwerem bazy danych, przetworzenia zapytania i zakończenia połączenia. Aby zmniejszyć związany z tym nakład pracy, produkty serwerowe DB2 Connect korzystają z techniki zestawiania połączeń do utrzymywania otwartych połączeń z bazą danych w łatwo dostępnej puli.

Większość aplikacji wykorzystujących technologię WWW wykonuje bardzo wiele krótkich transakcji. Typowa transakcja WWW jest wykonywana w ramach własnego połączenia. Innymi słowy, wykonanie transakcji oznacza nawiązanie połączenia z bazą danych i zakończenie tego połączenia po wykonaniu zaledwie kilku instrukcji SQL. Ten proces nawiązywania i kończenia połączeń jest bardzo kosztowny. Obejmuje utworzenie agenta DB2 Connect, nawiązanie połączenia sieciowego pomiędzy tym agentem i serwerem DB2 oraz utworzenie wątku DB2 na serwerze. W wypadku dłuższych połączeń koszty te rozkładają się na wszystkie transakcje wykonywane w ramach tego połączenia, ale dla typowej transakcji WWW przekraczają one koszty wykonania samej transakcji.

Tworzenie pul połączeń to technika, która umożliwia ponowne wykorzystanie infrastruktury nawiązanego połączenia dla kolejnych połączeń. Przy uruchamianiu instancji produktu DB2 Connect tworzona jest pula agentów koordynujących. Każdemu przychodzącemu połączeniu przypisywany jest agent. Nawiąże on połączenie z serwerem DB2 i w produkcie DB2 zostanie utworzony wątek. Gdy aplikacja wyśle żądanie rozłączenia, agent nie przekaże tego żądania do serwera DB2. Agent jest natomiast zwracany do puli. Agent w puli wciąż dysponuje swoim połączeniem z serwerem DB2 i odpowiadającym mu wątkiem DB2. Po zgłoszeniu żądania połączenia przez inną aplikację agent zostaje przypisany tej aplikacji. Aby zapewnić bezpieczeństwo operacji, dane o tożsamości użytkownika są przekazywane do wątku DB2, który z kolei przeprowadza proces uwierzytelniania użytkownika.

Pula połączeń DB2 zdecydowanie poprawia wydajność w takich środowiskach. Serwer DB2 Connect obsługuje otwarte połączenia z bazą danych w puli dostępnych połączeń. W momencie, gdy klient zgłasza żądanie połączenia, może ono być dostarczone z puli gotowych połączeń. Pula połączeń w znacznym stopniu zmniejsza nakład pracy poświęcany na otwieranie i zamykanie tych połączeń.

Technika zestawiania połączeń jest przezroczysta dla aplikacji łączących się z hostem poprzez produkt DB2 Connect. Gdy aplikacja żąda odłączenia od serwera, program DB2 Connect usuwa połączenie przychodzące dla aplikacji, ale utrzymuje w puli połączenie wychodzące dla hosta. Gdy nowa aplikacja żąda ustanowienia połączenia, program DB2 Connect używa

© Copyright IBM Corp. 1993, 2010

jednego z połączeń z puli. Użycie istniejącego połączenia skraca ogólny czas nawiązywania połączeń i zmniejsza wysoki koszt połączenia dla procesora w hoście.

Agenty DB2 Connect mogą być w jednym z dwóch stanów: w stanie bezczynności lub w stanie aktywnym. Agent jest w stanie aktywnym, gdy wykonuje pracę dla aplikacji. Po zakończeniu tej pracy agent przechodzi w stan bezczynności i oczekuje na dalszą pracę zleconą przez tę samą lub inną aplikację. Wszystkie bezczynne agenty zebrane są w tak zwanej puli bezczynnych agentów. Wielkość tej puli można określić za pomocą parametru konfiguracyjnego **num_poolagents**. Parametr ten jest równy maksymalnej liczbie bezczynnych agentów, jaką system ma obsługiwać. Nadanie temu parametrowi wartości zero oznacza wyłączenie opcji puli połączeń. Domyślną wartością tego parametru jest AUTOMATIC z wartością 100. Wartość AUTOMATIC powoduje, że program DB2 Connect automatycznie zarządza liczbą bezczynnych agentów w puli bezczynnych agentów.

Program DB2 Connect nie nawiązuje połączeń z bazą danych przed otrzymaniem pierwszego żądania od klienta. Ewentualnie można wypełnić pulę bezczynnych agentów zanim dowolny z klientów wygeneruje żądanie. Przy użyciu parametru konfiguracyjnego **num_initagents** pulę bezczynnych agentów można wypełnić przy uruchamianiu. Parametr ten określa, ile bezczynnych agentów powinno zostać utworzonych podczas uruchamiania. Początkowo nie będą one miały połączeń z serwerem bazy danych hosta.

Gdy klient zażąda połączenia z hostem, program DB2 Connect podejmie próbę uzyskania agenta spośród agentów znajdujących się w puli, którzy mają połączenie z serwerem bazy danych hosta. Jeśli próba ta się nie powiedzie, klient spróbuje znaleźć dostępnego agenta w puli bezczynnych agentów. Jeśli pula jest pusta, program DB2 Connect utworzy nowego agenta.

Można kontrolować maksymalną liczbę agentów, które mogą być równocześnie aktywne, używając w tym celu parametru konfiguracyjnego **max_coordagents**. Gdy liczba ta zostanie osiągnięta, nowe połączenie wygeneruje błąd o kodzie równym SQL1226. (Kod ten oznacza, że przekroczona została maksymalna liczba równoczesnych połączeń wychodzących). Domyślną wartością tego parametru jest AUTOMATIC z wartością 200. Wartość AUTOMATIC powoduje, że program DB2 Connect automatycznie zarządza liczbą agentów koordynujących.

Zmienna rejestrowa DB2CONNECT_IN_APP_PROCESS produktu DB2 powoduje, że aplikacje działające na tym samym komputerze, co produkt serwerowy DB2 Connect, mogą mieć uruchomiony produkt DB2 Connect w ramach procesu aplikacji, co jest domyślnym zachowaniem, lub mogą łączyć się z produktem serwerowym DB2 Connect i łączyć się z hostem dopiero w ramach agenta. Aby aplikacja mogła korzystać z pul połączeń, połączenia z hostem muszą być nawiązywane z poziomu agentów serwera DB2 Connect, a zatem parametr DB2CONNECT_IN_APP_PROCESS musi mieć wartość NO.

Pule połączeń DB2 Connect a pule połączeń serwera aplikacji

Pule połączeń są nieodzowne w wypadku aplikacji wykorzystujących technologie WWW i obsługujących duże liczby transakcji. Większość serwerów aplikacji WWW udostępnia obecnie własne metody grupowania połączeń z bazą danych w pule. Na przykład zarówno program Microsoft MTS (COM+), jak i IBM WebSphere udostępniają technikę zestawiania połączeń.

Aplikacyjne mechanizmy zestawiania połączeń implementowane przez te serwery w znaczący sposób różnią się od tego, który jest udostępniany przez serwery DB2 Connect. Ponieważ serwery aplikacji tworzą pule połączeń wyłącznie na własny użytek, przyjmują one zwykle, że identyfikator użytkownika, jego hasło oraz poziomy odseparowania będą dokładnie takie same dla wszystkich połączeń. Co ważniejsze, serwery aplikacji grupują tylko połączenia

zapoczątkowane przez ten sam proces. Oznacza to, że połączenia z innych komputerów, od innych użytkowników lub z innych procesów nie są włączane do puli. Choć takie techniki korzystania z pul połączeń przez serwery aplikacji są efektywne w przypadku ponownego wykorzystania połączeń nawiązanych przez tę samą instancję aplikacji, to są one całkowicie nieefektywne przy tworzeniu pul połączeń od wielu użytkowników lub z wielu serwerów.

Pule połączeń oferowane przez serwery DB2 Connect są całkowicie niezależne od aplikacji, komputera i użytkownika. Połączenia od wielu klientów, serwerów aplikacji z różnymi identyfikatorami użytkownika mogą ponownie wzajemnie wykorzystywać swoje połączenia, co w efekcie prowadzi do znacznie lepszego wykorzystania zasobów z puli.

Którego typu pul połączeń należy zatem używać? Obu. Na ogół użycie zarówno pul połączeń DB2 Connect, jak i serwerów aplikacji jest dobrą strategią, ponieważ nie kolidują one ze sobą. Nawet gdy na serwerze aplikacji włączona jest funkcja zestawiania połączeń, mechanizm zestawiania połączeń produktu DB2 Connect może udostępniać funkcję ponownego używania połączeń wielu serwerom aplikacji oraz innym klientom, korzystającym z serwera DB2 Connect.

Koncentrator połączeń

Koncentrator połączeń ogranicza zapotrzebowanie na zasoby na serwerach baz danych w systemie DB2 for OS/390 and z/OS przy obsłudze licznych stacji roboczych i użytkowników sieci WWW. Ta funkcja może radykalnie zwiększyć skalowalność systemu DB2 for OS/390 and z/OS i rozwiązania opartego o produkt DB2 Connect, udostępniając jednocześnie równoważenie obciążeń w środowiskach współużytkowania danych DB2 for OS/390 and z/OS na poziomie bezpiecznych operacji i transakcji.

Koncentrator połączeń umożliwia aplikacji utrzymywanie połączenia bez wykorzystywania żadnych zasobów serwera DB2 na hoście. Można mieć tysiące aktywnych użytkowników aplikacji i tylko kilka aktywnych wątków na serwerze DB2.

Technologia *koncentratora połączeń* produktu DB2 Connect pozwala produktom serwerowym DB2 Connect, takim jak DB2 Connect Enterprise Edition, zapewnić obsługę dla tysięcy użytkowników wykonujących jednocześnie transakcje biznesowe przy jednoczesnym radykalnym zmniejszeniu liczby wymaganych zasobów na hoście S/390 i serwerach bazy danych System i. Ten cel osiąga poprzez skoncentrowanie obciążeń ze wszystkich aplikacji w znacznie mniejszej liczbie połączeń hosta S/390 i serwera bazy danych System i. Chociaż mechanizm ten wygląda podobnie do opisanej powyżej funkcji puli połączeń, jest to jednak w rzeczywistości o wiele bardziej wysublimowane podejście do zmniejszania obciążenia zasobów przez aplikacje o wysokim poziomie przetwarzania transakcyjnego w trybie online (OLTP - Online Transaction Processing).

Koncentrator połączeń bazuje na koncepcji agenta, dzieląc go na dwie jednostki:

- Agent logiczny, który reprezentuje połączenie aplikacji.
- Agent koordynujący, który jest właścicielem połączenia DB2 i wątku oraz wykonuje żądania aplikacji.

Gdy nowa aplikacja próbuje nawiązać połączenie z hostem, przypisywany jest jej agent logiczny. Aby przekazać do bazy danych instrukcję SQL, niezbędny jest agent koordynujący, który zostaje przypisany w chwili inicjowania nowej transakcji. Kluczem do zrozumienia tej architektury jest fakt, że agent koordynujący jest:

- Rozdzielny z agentem logicznym.
- Zwracany do puli, gdy transakcja kończy się zatwierdzeniem lub wycofaniem zmian.

Innym kluczowym elementem jest metoda przypisywania agentów koordynujących do nowych transakcji w środowisku współużytkowania danych. Produkt DB2 Connect

implementuje zaawansowany algorytm planowania, korzystający z informacji dostarczanych przez środowisko Work Load Manager (WLM) systemów OS/390 i z/OS. Te informacje są używane do rozdzielania obciążeń wśród elementów grupy współużytkowania danych zgodnie z kryteriami ustawionymi w środowisku WLM. Menedżer WLM dysponuje nie tylko informacjami o obciążeniu każdego elementu grupy, ale także o jego dostępności. Umożliwia to produktowi DB2 Connect płynne przemieszczanie zadań z przeciążonych lub odciętych elementów grupy do działających elementów, wykorzystywanych dotąd w niewielkim stopniu. Koncentrator połączeń produktu DB2 Connect jest aktywowany w sytuacji, gdy ustawiona maksymalna liczba agentów logicznych (*max_connections*) jest wyższa niż liczba agentów koordynujących (*max_coordagents*).

Zestawianie połączeń zmniejsza koszty ustanawiania połączenia, gdy połączenie takie nie jest już potrzebne aplikacji kończącej działanie. Innymi słowy, jedna aplikacja musi się rozłączyć, zanim inna będzie mogła skorzystać z połączenia znajdującego się w puli połączeń.

Ewentualnie koncentrator połączeń umożliwia produktowi DB2 Connect udostępnienie aplikacji połączenia, gdy tylko inna aplikacja zakończy transakcję, nie wymagając jej rozłączenia się. Zasadniczo połączenie serwera bazy danych oraz skojarzony z nim host i zasoby produktu DB2 Connect są używane przez aplikację tylko w trakcie wykonywania transakcji. Natychmiast po zakończeniu transakcji połączenie oraz skojarzone z nim zasoby mogą być wykorzystane przez inne aplikacje, które muszą wykonać transakcję.

We wcześniejszych wersjach produktu DB2 Connect każda aktywna aplikacja miała jednostkę rozsyłalną mechanizmu (Engine Dispatchable Unit - EDU), która zarządzała połączeniem z bazą danych oraz żądaniami aplikacji. Taka jednostka EDU nazywana była zwykle *agentem koordynacji*. Każdy agent koordynacji śledził stan lub kontekst aplikacji i jednostki EDU. Jeśli liczba połączeń wzrastała, każda jednostka EDU zajmowała coraz większą ilość pamięci, a ponadto przełączanie kontekstu między agentami powodowało powstawanie dodatkowego narzutu.

W przedstawionej powyżej architekturze, między połączeniami i jednostkami EDU istnieje związek jeden do jednego. Jednak koncentrator połączeń pozwala na powstawanie między połączeniami i jednostkami EDU związku jeden do wielu. Oznacza to, że związki między połączeniami (X) i jednostkami EDU (Y) można przedstawić teraz jako X >= Y.

Koncentrator połączeń dzieli agenta na dwie jednostki: *agenta logicznego* i *agenta pracującego*. Agenty logiczne reprezentują aplikację, lecz nie mają odwołania do konkretnej jednostki EDU. Agent logiczny zawiera wszystkie informacje i bloki kontrolne wymagane przez aplikację. Jeśli jest *n* aplikacji połączonych z serwerem, to na serwerze musi istnieć *n* agentów logicznych. Agent pracujący to fizyczna jednostka EDU, która wykonuje żądania aplikacji, lecz która nie ma trwałego połączenia z daną aplikacją. Do wykonywania transakcji agenty pracujące tworzą powiązania z agentami logicznymi, a granice transakcji kończą powiązania i zwracają je do dostępnej puli.

Jednostka nazywana *programem rozsyłającym* przypisuje agenty pracujące do agentów logicznych. Ograniczenie liczby otwartych uchwytów plików, obecne na niektórych platformach komputerowych, może powodować tworzenie wielu instancji programu planującego.

Ograniczenia koncentratora połączeń

Istnieje pewna liczba istotnych ograniczeń dotyczących korzystania z koncentratora serwera DB2 Connect. Przed podjęciem próby użycia koncentratora we własnym systemie, należy przeczytać wszystkie poniższe informacje.

Ograniczenia ogólne:

- Do nawiązywania połączeń przychodzących od zdalnych i lokalnych klientów koncentrator wykorzystuje protokół TCP/IP. Z wychodzących połączeń znajdujących się w puli mogą korzystać tylko połączenia przychodzące wykorzystujące z protokół TCP/IP lub połączenia przychodzące lokalne (IPC). Koncentrator zaakceptuje połączenia za pośrednictwem innych protokołów komunikacyjnych, takich jak potoki nazwane, lecz nie będzie mógł użyć do tych połączeń swoich opcji koncentrowania XA.
- W celu obsługi transakcji ściśle powiązanych z interfejsem XA wszystkie aplikacje uczestniczące w tej samej transakcji XA muszą używać tej samej instancji serwera DB2 Connect do łączenia się z hostem.
- Jedynie aplikacje, które zamykają wstrzymane zasoby (takie jak wstrzymane kursory) przy osiągnięciu granicy transakcji, mogą czerpać korzyści z użycia koncentratora. Transakcje, które nie zamkną wstrzymanych kursorów przejdą dalej, lecz będą powiązane z dedykowanym agentem pracującym i dlatego też nie będą mogły korzystać z pełnego zbioru opcji koncentratora.
- Jeśli zostaną zadeklarowane globalne tabele tymczasowe, muszą być one zamknięte w sposób jawny na granicy transakcji lub gałęzi. Jeśli tabele nie zostaną zamknięte, koncentrator połączeń zostanie wyłączony, ale aplikacja będzie nadal działać.
- Wszystkie aplikacje biorące udział w tej samej transakcji XA muszą mieć ten sam identyfikator CCSID i muszą używać tej samej nazwy użytkownika w celu wykonania połączenia.
- Jeśli połączenie wychodzące zostało ustanowione w celu obsługi transakcji zatwierdzania dwufazowego, agent takiego połączenia może być użyty tylko do obsługi połączeń dwufazowych. Podobnie agenty ustanowione do obsługi jednofazowych połączeń mogą obsługiwać tylko połączenia jednofazowe.
- Koncentrator obsługuje aplikacje korzystające ze sterownika IBM Data Server Driver for JDBC and SQLJ oraz aplikacje CLI, które korzystają z dynamicznego SQL. Aplikacje wykorzystujące interfejs CLI nie powinny także używać instrukcji KEEPDYNAMIC, ponieważ koncentrator jest zależny od instrukcji ponownie przygotowywanych po zakończeniu każdej transakcji.
- Żądania przygotowane w sposób dynamiczny z aplikacji utworzonych we wbudowanym języku SQL zostaną odrzucone. W związku z tym aplikacje powinny być zmienione w taki sposób, aby używały statycznego języka SQL albo dla dynamicznych instrukcji SQL używały interfejsu CLI.

Podczas pracy z produktem DB2 w wersji 9 lub 8 z pakietem poprawek 13 (lub nowszym) do włączenia obsługi koncentratora produktu DB2 Connect wymagany jest produkt System i w wersji 5 wydanie 4 (PTF SI23726). W przeciwnym razie obsługiwana jest jedynie część interfejsu XA koncentratora połączeń.

Aktywacja koncentratora połączeń

Parametr konfiguracyjny menedżera bazy danych *max_coordagents* określa maksymalną liczbę agentów logicznych. Opcja koncentratora zostaje włączona przez zwiększenie wartości parametru *max_connections* powyżej wartości domyślnej. Domyślna wartość parametru *max_connections* jest równa wartości parametru *max_coordagents*. Ponieważ każda aplikacja będzie miała jednego agenta logicznego, parametr *max_connections* faktycznie decyduje o liczbie aplikacji, które mogą być podłączone do instancji bazy danych, natomiast parametr *max_coordagents* decyduje o liczbie połączeń przychodzących, które mogą być jednocześnie aktywne. Wartości parametru *max_connections* mogą należeć do zakresu od *max_coordagents* do 64 000. Domyślna liczba agentów logicznych jest równa *max_coordagents*.

Zarówno dla parametru *max_connections*, jak i parametru *max_coordagents* można ustawić opcję AUTOMATIC. Jeśli opcja AUTOMATIC zostanie ustawiona dla parametru

max_connections, liczba połączeń może zostać zwiększona ponad skonfigurowaną wartość bazową. Jeśli opcja AUTOMATIC zostanie ustawiona zarówno dla parametru *max_connections*, jak i parametru *max_coordagents*, parametr *max_connections* może zostać zwiększony ponad wartość bazową, natomiast wartość parametru *max_coordagents* będzie zwiększana automatycznie w celu zachowania współczynnika koncentracji między połączeniami a agentami koordynacji.

Do konfigurowania agentów używa się kilku istniejących parametrów konfiguracyjnych. Są to następujące parametry:

max_coordagents
: Maksymalna liczba aktywnych agentów koordynujących.

num_poolagents
: Wielkość puli agentów. Pula agentów składa się z agentów nieaktywnych i z agentów bezczynnych. W celu uzyskania lepszej wydajności parametr *num_poolagents* powinien być skonfigurowany w taki sposób, aby miał wartość równą średniej liczbie klientów.

num_initagents
: Początkowa liczba agentów pracujących znajdujących się w puli. Mogą to być agenty bezczynne.

obsługa transakcji XA

Architektura koncentratora połączeń umożliwia produktowi DB2 Connect udostępnianie obsługi transakcji ściśle powiązanych z interfejsem XA dla produktu DB2 for OS/390 and z/OS i DB2 for System i. Koncentrator połączy agenta pracującego z określoną transakcją XA (pojedynczym identyfikator XID) tak, jak w wypadku każdej innej transakcji. Jeśli jednak transakcja XA kończy się przez xa_end() (granica gałęzi), agent pracujący nie zostanie zwrócony do puli ogólnej. Zamiast tego agent pracujący pozostanie stowarzyszony z tą konkretną transakcją XA. Gdy inna aplikacja przyłączy się do tej samej transakcji XA, agent pracujący zostanie podłączony do tej aplikacji.

Każde wywołanie granicy transakcji zwróci agenta do puli. Na przykład wywołania xa_prepare() w trybie tylko do odczytu, xa_rollback(), xa_recover(), xa_forget(), xa_commit() lub dowolny błąd XA, który spowoduje wycofanie zmian, zwrócą agenta do zwykłej puli. Samo wywołanie xa_end() kończy tylko gałąź transakcji i nie wystarczy, aby zakończyć powiązanie transakcji z identyfikatorem XID.

Przykłady obsługi transakcji XA

1. Wyobraźmy sobie środowisko, w którym potrzebnych jest 4000 lub więcej jednoczesnych połączeń. Taka sytuacja może występować na serwerze sieci WWW, który korzysta z aplikacji CGI lub w systemie biurowym, do którego podłączonych jest wielu użytkowników. W takich przypadkach względy wydajności wymagają zwykle, aby program DB2 Connect pracował jako autonomiczny gateway, co oznacza, że baza danych i system DB2 Connect muszą znajdować się na oddzielnych komputerach.

   System serwera DB2 Connect może nie być w stanie obsłużyć równocześnie 4000 otwartych połączeń z komputerem bazy danych. W większości wypadków liczba transakcji przeprowadzanych w danym momencie będzie znacznie mniejsza niż liczba połączeń współbieżnych. Administrator systemu może wtedy zwiększyć wydajność systemu, ustawiając parametry konfiguracyjne bazy danych w następujący sposób:

   ```
   MAX_CONNECTIONS =  4000
   MAX_COORDAGENTS =  1000
   NUM_POOLAGENTS  =  1000
   ```

Koncentrator może utrzymać do 4000 otwartych, jednocześnie działających sesji, nawet jeśli gateway może równocześnie obsługiwać tylko 1000 transakcji.

2. W powyższym przykładzie agenty pracujące będą nieustannie tworzyć i usuwać powiązania z agentami logicznymi. Te agenty, które nie są bezczynne, mogą obsługiwać połączenia z bazą danych, lecz nie biorą udziału w żadnej określonej transakcji, dlatego też są dostępne dla dowolnego agenta logicznego (aplikacji), który żąda połączenia.

   Przypadek dotyczący transakcji XA jest trochę inny. W tym przykładzie zakłada się, że monitor przetwarzania transakcyjnego jest używany z gatewayem produktu DB2 Connect i bazą danych zSeries lub System i. Gdy aplikacja żąda połączenia, koncentrator może odwrócić stan nieaktywnego agenta, aby ten obsłużył żądanie, lub utworzyć nowego agenta pracującego. Załóżmy, że aplikacja żąda transakcji XA. Dla tej transakcji tworzony jest identyfikator XID i zostaje z nim powiązany agent pracujący.

   Gdy żądanie aplikacji zostanie obsłużone, wywołuje ona funkcję xa_end() i odłącza się od agenta pracującego. Agent pracujący pozostaje powiązany z identyfikatorem XID transakcji. Może on teraz obsługiwać tylko żądania transakcji z powiązanym z nim identyfikatorem XID.

   W tym momencie inna aplikacja może zgłosić żądanie transakcji innej niż XA. Nawet jeśli nie będzie żadnych innych wolnych agentów pracujących, agent powiązany z identyfikatorem XID nie będzie dostępny dla drugiej aplikacji. Uważany jest on za agenta aktywnego. Dla drugiej aplikacji musi zostać utworzony nowy agent pracujący. Gdy druga aplikacja zakończy swoją transakcję, związany z nią agent pracujący zostanie zwrócony do puli dostępnych agentów.

   W międzyczasie inne aplikacje żądające transakcji powiązanych z identyfikatorem XID pierwszego agenta mogą podłączać się do tego agenta oraz odłączać się od niego, co spowoduje wykonanie dla nich dedykowanej transakcji XA. Każda aplikacja żądająca tej konkretnej transakcji zostanie wysłana do tego agenta pracującego, jeśli będzie on wolny.

   Agent pracujący nie zostanie zwolniony do puli ogólnej, dopóki aplikacja nie zgłosi wywołania granicy transakcji (nie xa_end()). Na przykład aplikacja może zakończyć transakcję wywołaniem xa_commit() i w tym miejscu agent pracujący usunie powiązanie z identyfikatorem XID i wróci do puli dostępnych agentów. Każda aplikacja zgłaszająca żądanie może wtedy użyć go zarówno do innej transakcji XA, jak też do transakcji innego typu niż XA.

Pule połączeń i koncentrator połączeń

Chociaż pule połączeń i koncentrator połączeń wydają się wykazywać pewne podobieństwa, mechanizmy te różnią się w implementacji i służą różnym celom. Pule połączeń pomagają ograniczyć nakład pracy związany z obsługą połączeń z bazą danych i obsłużyć znaczną ilość połączeń. Koncentrator połączeń pomaga w zwiększeniu skalowalności opartego o system DB2 for OS/390 and z/OS i produkt DB2 Connect rozwiązania przez optymalizowanie użycia serwerów bazy danych hosta.

Połączenie należące do puli połączeń może być ponownie wykorzystane tylko wtedy, gdy aplikacja, do której należy połączenie, zgłosi żądanie rozłączenia. W wielu dwuwarstwowych aplikacjach typu klient/serwer użytkownicy nie rozłączają się w ogóle w ciągu całego dnia roboczego. Podobnie większość serwerów aplikacji w aplikacjach wielowarstwowych nawiązuje połączenia z bazą danych przy uruchamianiu serwera i nie zwalnia ich aż do zatrzymania serwera aplikacji.

W tych środowiskach zestawianie połączeń przyniesie co najwyżej niewielką korzyść. Jednak w środowiskach WWW i środowiskach klient/serwer, w których częstotliwość połączeń i rozłączeń jest wyższa, pule połączeń spowodują znaczną poprawę wydajności. Koncentrator połączeń przydziela zasoby bazy danych na hoście tylko na czas trwania transakcji SQL, utrzymując jednocześnie aktywność aplikacji użytkownika. Umożliwia to tworzenie

konfiguracji, w których liczba wątków programu DB2 oraz ilość wykorzystywanych przez nie zasobów może być znacznie mniejsza niż w sytuacji, w której każda aplikacja miałaby swój własny wątek.

Jeśli chodzi o bezawaryjne działanie i równoważenie obciążenia, koncentrator połączeń jest wyborem oczywistym, gdyż umożliwia on przenoszenie obciążenia z każdą nową transakcją. Technika zestawiania połączeń może oferować jedynie bardzo ograniczone równoważenie i to tylko w trakcie połączenia.

Z uwagi na różnice w ich przeznaczeniu mechanizmy pul połączeń i koncentratora połączeń powinny być używane jednocześnie.

Menedżer transakcji WebSphere MQ i produkt DB2 for OS/390 wymagają koncentratora połączeń

W przypadku uruchamiania aplikacji w środowisku IBM WebSphere MQ (znanym wcześniej jako IBM MQSeries), produkt WebSphere MQ może działać jako menedżer transakcji zgodny z interfejsem XA, koordynujący wszystkie rozproszone transakcje zatwierdzania dwufazowego. Jeśli produkt WebSphere MQ działa jako taki menedżer transakcji, a źródła danych to rodzina produktów DB2, należy spełnić określone wymagania konfiguracyjne.

Większość wymagań konfiguracyjnych dla takiego środowiska menedżera transakcji została już opisana w innych miejscach. Na przykład, parametr konfiguracyjny DB2 TP_MON_NAME w produkcie DB2 Runtime Client musi mieć wartość “MQ”.

Jednak jedno wymaganie konfiguracyjne zostało pominięte. Wymaganie to dotyczy programu DB2 Connect, który łączy się ze źródłami danych w postaci serwerów DB2 for OS/390. Aby używać produktu WebSphere MQ do koordynowania rozproszonych transakcji, w których uczestniczą serwery DB2 for z/OS i DB2 for i5/OS, należy włączyć opcję koncentratora połączeń DB2 Connect na bramie. Koncentrator połączeń jest włączony, jeśli wartość parametru **max_connections** jest większa od wartości parametru **max_coordagents**.

Jeśli koncentrator połączeń nie zostanie włączony, może to spowodować nieprzewidziane zachowanie transakcji.

Rozdział 18. Obsługa systemu Sysplex w serwerach DB2 Connect

System Sysplex to kolekcja serwerów System z współdziałających ze sobą na poziomie sprzętu i oprogramowania w celu wykonywania zadań. Sysplex koordynuje współpracę poprzez zwiększanie liczby pracujących równocześnie procesorów, co zwiększa liczbę zadań, które mogą zostać wykonane. Poza wzrostem możliwości przetwarzania system Sysplex zapewnia elastyczność łączenia poziomów sprzętu i oprogramowania oraz dynamicznego dodawania systemów.

Sysplex umożliwia serwerowi DB2 Connect bezproblemowe równoważenie połączeń wśród elementów grupy współużytkującej dane. System Sysplex udostępnia także serwerowi DB2 Connect mechanizm wyboru alternatywnych elementów w przypadku awarii jednego z elementów. Możliwość przekierowywania w systemie Sysplex jest funkcją programu DB2 Connect. Obsługa systemu Sysplex dla serwerów DB2 Connect jest domyślnie włączona, podobnie jak możliwość przekierowywania w systemie Sysplex. Obsługę systemu Sysplex w bazie danych na systemie hostowym można wyłączyć, usuwając parametr SYSPLEX z jej pozycji katalogu DCS, jednak nie należy usuwać samej pozycji DCS, nawet gdy nie jest w niej określony żaden parametr.

Przy automatycznym przekierowywaniu klienta dla połączenia z włączoną obsługą systemu Sysplex domyślnym zachowaniem w razie niepowodzenia komunikacji jest ponowienie próby nawiązania połączenia. Kiedy program DB2 Connect jest połączony z serwerem DB2 for z/OS, wartości rejestru specjalnego są odtwarzane aż do ostatniej pomyślnej transakcji nie wstrzymującej zasobów.

Można precyzyjnie skonfigurować zachowanie związane z ponawianiem operacji automatycznego przekierowywania klientów, łącznie z możliwością wyłączenia tej funkcji, korzystając ze zmiennych rejestrowych **DB2_MAX_CLIENT_CONNRETRIES** i **DB2_CONNRETRIES_INTERVAL**. Zmienna rejestrowa limitu czasu połączenia to **DB2TCP_CLIENT_CONTIMEOUT**.

Uwagi dotyczące wykorzystania systemu SYSPLEX w systemach OS/390 i zSeries

Produkt DB2 Connect przy kierowaniu połączeń do wielu systemów Sysplex zapewnia równoważenie obciążenia i odporność na błędy. Przy połączeniu z serwerem bazy danych DB2 for OS/390 and z/OS działającym w środowisku współużytkowania danych produkt DB2 Connect w oparciu o dostarczane przez program Workload Manager (WLM) dane o obciążeniu systemu rozdzieli obciążenie na różne podsystemy DB2 składające się na grupę współużytkowania danych.

Produkt DB2 Connect otrzymuje z programu WLM uszeregowaną według priorytetu listę elementów systemu Sysplex. Każdy system Sysplex zwraca ważoną informację o priorytecie dla każdego adresu połączenia. Ta lista jest następnie używana przez produkt DB2 Connect do rozdzielania przychodzących żądań CONNECT pomiędzy elementami systemu Sysplex z przypisanymi najwyższymi priorytetami. Na potrzeby funkcji równoważenia obciążenia lista ważonych informacji o priorytetach Sysplex jest uzyskiwana podczas każdego połączenia. Jeśli włączony jest koncentrator połączeń produktu DB2 Connect, ta lista jest używana także przy podejmowaniu decyzji, dokąd wysłać każdą transakcję.

© Copyright IBM Corp. 1993, 2010

Uwaga: Aby skorzystać z obsługi systemu Sysplex w produkcie DB2 Connect, nie trzeba dokonywać żadnych zmian w konfiguracji narzędzia obsługi danych rozproszonych (DDF) OS/390 i z/OS.

Produkt DB2 Connect zapewnia również odporność na błędy przez podejmowanie prób połączenia z alternatywnym komputerem systemu Sysplex w przypadku awarii połączenia. Tylko w sytuacji, gdy żadne ze znanych połączeń nie będzie dostępne, do aplikacji zostanie zwrócony błąd.

Obsługa systemu Sysplex w produkcie DB2 Connect została zaprojektowana z myślą o grupowaniu agentów w pule. Przy włączonej funkcji Sysplex w przypadku, gdy połączenie z elementem DDF zostanie zerwane, produkt DB2 Connect kieruje połączenia do innego elementu DDF. Przekierowanie jest dokonywane w oparciu o listę serwerów Sysplex.

Wraz z koncentratorem produkt DB2 Connect ma teraz możliwość równoważenia obciążenia w granicach transakcji. Aby ta funkcja działała, koncentrator produktu DB2 Connect musi być włączony.

Wykorzystanie systemu Sysplex w DB2

W typowym scenariuszu serwer DB2 Connect (serwer A) wymienia informacje z systemem Sysplex obejmującym dwa serwery DB2 for OS/390 and z/OS (serwery B i C).

Serwer B systemu Sysplex	Serwer C systemu Sysplex
HOST_NAME=MVSHOST	HOST_NAME=MVSHOST1

Załóżmy, że w tym scenariuszu aplikacja wydaje komendę:

```
db2 connect to aliasb user xxxxxxx using xxxxxxxx
```

Nawiązane zostaje połączenie z bazą danych na serwerze MVSHOST. Ponieważ użycie systemu Sysplex jest włączone zarówno na serwerze DB2 Connect, jak i w pozycji katalogu DCS, produkt DB2 for OS/390 and z/OS identyfikuje adresy sieciowe do produktu DB2 Connect dla każdego elementu systemu Sysplex. (Do przekazania tych informacji używane są protokoły DRDA4 i przepływy komunikatów pomiędzy serwerami MVSHOST i MVSHOST1). Gdy pierwsze połączenie zostanie nawiązane, zwracana lista adresów zostaje zapisana w pamięci podręcznej na stacji roboczej DB2 Connect. Gdy tylko początkowe żądanie CONNECT zostanie wysłane do węzła TCP/IP, zwracane są adresy IP.

Dane o priorytetach zapewniające równoważenie obciążenia i odporność na błędy

Lista adresów udostępniana przez produkt DB2 for OS/390 and z/OS zawiera również informacje o priorytetach, w tym liczbę połączeń dla każdego adresu sieciowego. Ta lista jest odświeżana za każdym razem, gdy produkt DB2 Connect tworzy nowe połączenie. Ta dodatkowa informacja jest używana na potrzeby równoważenia obciążenia, jak również w celu zapewnienia odporności na błędy.

Lista adresów w pamięci podręcznej używana przez produkt DB2 Connect

Jeśli połączenie bazy danych do ALIASB ulega awarii, generowany jest komunikat o błędzie SQL30081N i połączenie zostaje zerwane. Jeśli odebrane zostanie następne żądanie połączenia do ALIASB, produkt DB2 Connect wykona następujące czynności:

1. Spróbuje wykorzystać serwer o najwyższym priorytecie z zapamiętanej w pamięci podręcznej listy adresów opartej na informacji o priorytetach zwracanej przez produkt DB2 for OS/390 and z/OS. Produkt DB2 Connect zawsze używa tej strategii, dzięki czemu zapewnia równoważenie obciążenia.
2. Jeśli ta próba połączenia nie powiedzie się, sprawdzone zostaną kolejne adresy z listy w malejącej kolejności priorytetów zwróconych przez produkt DB2 for OS/390 and z/OS. W ten sposób produkt DB2 Connect wykorzystuje dane Sysplex, aby uzyskać odporność na błędy.
3. Jeśli żadna z kolejnych prób połączenia nie powiedzie się, produkt DB2 Connect spróbuje ponownie nawiązać połączenie do ALIASB, stosując adres zawarty w katalogu węzłów.

Komenda db2pd z parametrem sysplex (db2pd -sysplex) może zostać zastosowana do odtworzenia danych o serwerach skojarzonych z systemem Sysplex.

Wymagania konfiguracyjne systemu Sysplex

Wykorzystanie systemu Sysplex nie będzie dla danej bazy danych możliwe, chyba że pozycja w katalogu DCS dla tej bazy danych zawiera słowo Sysplex (bez rozróżniania wielkości liter) w miejscu szóstego w kolejności parametru.

Rozdział 19. Dostrajanie programu DB2 Connect

Do dostrajania programu DB2 Connect mogą zostać użyte różne parametry z pliku konfiguracyjnego menedżera bazy danych.

RQRIOBLK

Parametr **RQRIOBLK** ustawia maksymalną wielkość sieciowych bloków we/wy. Użycie większego bloku może poprawić wydajność przesyłania większych żądań. Wielkość bloku nie ma zazwyczaj wpływu na czas odpowiedzi mniejszych żądań, na przykład żądania dotyczącego pojedynczego wiersza danych.

Większe bloki wymagają zwykle więcej pamięci na serwerze DB2 Connect. Powoduje to zwiększenie zajętego obszaru pamięci i może zwiększyć liczbę operacji stronicowania na małych stacjach roboczych.

Należy używać domyślnej wielkości bloku w architekturze DRDA (32767), o ile nie spowoduje to zbyt intensywnego stronicowania przy wykonywaniu aplikacji. W przeciwnym wypadku należy zmniejszać wielkość bloku we/wy, aż do zlikwidowania stronicowania. Po rozpoczęciu stronicowania nastąpi zauważalne zmniejszenie wydajności. Przy pomocy narzędzi do monitorowania wydajności (takich jak narzędzie vmstat dla systemów operacyjnych Linux i UNIX) można sprawdzić, czy w systemie występuje stronicowanie.

DIR_CACHE

Parametr **DIR_CACHE** określa, czy informacje dotyczące katalogu są zapisywane w pamięci podręcznej. Przy użyciu pamięci podręcznej (DIR_CACHE=YES) pliki katalogu są odczytywane i zapisywane w pamięci podręcznej w celu zminimalizowania nakładu pracy tworzenia wewnętrznej struktury katalogu i odczytywania plików katalogu przy każdym nawiązaniu połączenia.

Bez używania pamięci podręcznej (DIR_CACHE=NO) przy każdym połączeniu z bazą danych z dysku odczytywany jest odpowiedni katalog i wykonywane jest przeszukiwanie. Po znalezieniu żądanych pozycji zwalniana jest cała pamięć związana z przeszukiwaniem katalogu.

Przy włączonym buforowaniu pamięć podręczna współużytkowanego katalogu jest budowana podczas przetwarzania komendy db2start i zwalniana po zatrzymaniu działania programu DB2. Pamięć ta jest używana przez wszystkie procesy serwera DB2 (db2agent). Również pamięć podręczna katalogu prywatnej aplikacji jest tworzona przy pierwszym odwołaniu aplikacji do bazy danych i jest zwalniana po zakończeniu aplikacji.

Każda pamięć podręczna dostarcza obraz systemowego katalogu baz danych, katalogu DCS i katalogu węzłów. Pamięć podręczna redukuje koszty związane z połączeniem przez eliminowanie operacji we/wy pliku katalogu i minimalizowanie przeszukiwania katalogu.

Po zaktualizowaniu zmiany nie są natychmiast wprowadzane do pamięci podręcznej. Jeśli w pamięci podręcznej nie zostanie odnaleziona pozycja katalogu, przeszukiwany jest katalog pierwotny.

Pamięć podręczna zwiększa pamięć prywatną potrzebną do wykonywania aplikacji. Jeśli pamięć podręczna nie istnieje, pamięć prywatna jest używana do przeszukiwania katalogu. Ogólnie użycie pamięci współużytkowanej przez program DB2 wzrasta nieznacznie,

© Copyright IBM Corp. 1993, 2010

ponieważ informacje z katalogu współużytkowane przez agenty baz danych są przesyłane do tej pamięci. Wymagana wielkość pamięci podręcznej zależy od liczby pozycji zdefiniowanych w każdym katalogu.

NUMDB

Wartość parametru konfiguracyjnego **NUMDB** nie miała wpływu na działanie poprzednich wersji produktu DB2 Connect, zostało to jednak zmienione począwszy od wersji 8. Parametr ten wskazuje maksymalną liczbę baz danych, do których klienty mogą uzyskać połączenia za pośrednictwem serwera DB2 Connect. Dokładniej, jest to maksymalna liczba różnych aliasów baz danych, które mogą zostać wpisane do katalogu na serwerze DB2 Connect.

Inne parametry programu DB2 Connect

Parametry **AGENTPRI** i **MAXAGENTS** są nieaktualne w wersji 9.5

Komendy aktualizujące wartości parametru **MAXAGENTS** będą nadal działały, aby zapewnić działanie istniejących aplikacji, ale wartości te będą ignorowane. Nazwa parametru nie występuje na żadnej z list konfiguracyjnych. W poprzednich wersjach łączna liczba agentów tworzonych w danej partycji DB2 była kontrolowana za pomocą parametru **MAXAGENTS**. Teraz istnieje możliwość zautomatyzowania procesu konfiguracji agentów.

Domyślnie, parametr **NUM_POOLAGENTS** zostanie ustawiony na AUTOMATIC z domyślną wartością 100. Również parametr **MAX_COORDAGENTS** zostanie domyślnie ustawiony na AUTOMATIC z domyślną wartością 200.

Aby przesłać łańcuchy rozliczeniowe z aplikacji klienta do serwera DB2 Connect, należy użyć właściwych dla interfejsu API metod ustawiania informacji rozliczeniowej. Metody właściwe dla interfejsu API działają szybciej niż ustawienie zmiennej środowiskowej DB2ACCOUNT.

IBM Data Server Driver for JDBC and SQLJ
Właściwość com.ibm.db2.jcc.DB2BaseDataSource.clientAccountingInformation

Dostawca danych DB2 dla platformy .NET
Właściwość DB2Connection.ClientAccountingInformation

CLI/ODBC
Parametr konfiguracyjny ClientAcctStr CLI/ODBC

Wbudowany SQL (C, C++ i COBOL)
Funkcja sqlesact

Jeśli nie jest potrzebny dopasowany plik odwzorowujący SQLCODE, wydajność można poprawić przez używanie domyślnego odwzorowania SQLCODE lub wyłączenie go. Domyślny plik odwzorowania jest wbudowany w bibliotekę DB2 Connect; dostosowany plik odwzorowania musi być wczytany z dysku, co niekorzystnie wpływa na wydajność.

Dostrajanie bazy danych hosta

Na wydajność systemu będzie miała wpływ wydajność hosta lub serwera bazy danych System i. Różne systemy zarządzania baz danych mają różne opcje wydajności. Optymalizatory języka SQL w różnych systemach mogą zachowywać się w różny sposób, nawet jeśli używana jest ta sama aplikacja. Dodatkowe informacje na ten temat można znaleźć w dokumentacji dotyczącej wydajności systemu serwera baz danych na hoście lub serwerze System i.

Warto poprawić wydajność przy użyciu opcji wiązania z odczytem niezatwierdzonym (UR) lub bez zatwierdzenia (NC) wszędzie, gdzie jest to możliwe, aby uniknąć chronologicznego zapisywania zmian.

Uwaga: W przypadku użycia odczytu niezatwierdzonego (UR) dane niezapisane do kroniki można tylko odczytywać, ale nie można ich aktualizować i można to robić tylko w przypadku, gdy opcja łączenia danych w bloki ma wartość ALL.

W zależności od serwera aplikacji i dostarczonej przez niego granulacji blokowania, poziom wyodrębnienia używany dla zapytania lub aplikacji może mieć znaczący wpływ na wydajność. Baza danych powinna mieć odpowiedni poziom normalizacji, skuteczne wykorzystanie indeksów i odpowiednio przydzielaną przestrzeń bazy danych. Na wydajność mogą mieć również wpływ typy danych, co zostało opisane poniżej.

Uwagi dotyczące dostrajania sieci

Najlepszym sposobem zwiększenia ogólnej wydajności w środowisku rozproszonej bazy danych jest usunięcie opóźnień powstających w sieci. Administratorzy sieci często zauważają, że praca w sieci jest wydajniejsza, jeśli między transmisjami zostaje zgromadzonych tak dużo danych, jak to tylko możliwe. Tego podejścia nie da się zastosować dla aplikacji, takich jak rozproszone bazy danych, ponieważ to one generują opóźnienia w sieci. Użytkownik końcowy nie dostrzega wydajności sieci, jedynie opóźnienia.

Większość urządzeń sieciowych ma parametry opóźnienia i ustawione wartości domyślne, które są nieodpowiednie dla rozproszonych baz danych. Aby zwiększyć wydajność, należy odnaleźć te parametry i, o ile jest to możliwe, ustawić dla nich wartość zero. Ponadto należy sprawdzić, czy wielkość buforu urządzenia jest wystarczająca, aby zapobiec retransmisjom spowodowanym przez utratę danych. Na przykład systemy UNIX zazwyczaj mają domyślną wartość głębokości kolejek transmisji i odbierania równą 32. Aby zwiększyć wydajność, należy ustawić głębokość kolejki na 150. W ustawieniach sterowania łączem danych odpowiednim parametrem jest głębokość wejściowa, która również powinna wynosić 150.

W większości miejsc parametr IOBUF ma za małą wartość. Ma on zwykle wartość 500, ale doświadczenie pokazuje, że wartość 3992 powoduje najlepszą pracę, gdy przekazywane są duże ilości danych, szczególnie dla połączeń poprzez kanał, takich jak ESCON lub 3172.

W systemie LAN wielkości okien transmisji i odbierania DLC (sterowanie łączem danych) lub LLC (sterowanie łączem logicznym) mogą mieć ogromny wpływ na wydajność. Wartość wysyłania powinna mieć wartość 7 lub większą. Dla większości konfiguracji najlepsza wartość odbioru wynosi 4 lub mniej.

Jeśli używana jest sieć Ethernet, należy ustawić wielkość segmentu TCP na 1500 bajtów. Dla sieci Token Ring lub FDDI wartość ta powinna być ustawiona na 4400 bajtów, a w przypadku korzystania z adaptera ESCON z protokołem TCP/IP, wielkość segmentu powinna zawsze mieć wartość 4096.

Dla sieci TCP/IP wielkości buforów wysyłania i odbioru TCP powinny być ustawione na wartość większą niż 32768. Ogólnie najlepszą wartością jest 65536.

Uwaga: Ustanowienie połączenia od gatewaya do serwera (połączenie wychodzące) jest znacznie kosztowniejsze niż ustanowienie połączenia od klienta do gatewaya (połączenie przychodzące). W środowisku, w którym tysiące klientów często łączy (i rozłącza) się z serwerem przez gateway, ustanawianie połączeń wychodzących stanowi znaczną część czasu przetwarzania. Program DB2 Connect umożliwia kolejkowanie połączeń przez protokół TCP/IP. Gdy klient żąda odłączenia od serwera, gateway usuwa połączenie przychodzące z klientem, ale utrzymuje w puli połączenie wychodzące z serwerem. Gdy nowy klient pojawia

się w gatewayu, żądając połączenia, gateway udostępnia istniejące połączenie z puli, ograniczając w ten sposób całkowity czas połączenia i redukując wysoki koszt połączenia CPU na serwerze.

Podsumowanie metod dostrajania wydajności sieci zawarto w tabeli Tabela 17.

Tabela 17. Metody dostrajania wydajności sieci

Czego należy szukać	Przykład	Ustawienia	Uwagi
Celowe opóźnienia	Parametry opóźnienia dla urządzeń sieciowych	Ustawić na 0.	Ustawienia domyślne mają zwykle większe wartości.
Bufory	Parametr IOBUF	Ustawić na 3992.	Szczególnie korzystne dla adaptera ESCON lub innego adaptera kanału.
Bufory	RUSIZE	Optymalna wielkość to 4096.	Ustawienie parametrów RUSIZE i RQRIOBLK na tę samą wielkość może dać najlepszą wydajność.
Bufory	Pacing	VPACING, PACING i profile trybu należy ustawić na 63.	Jeśli można, należy zastosować pacing dostosowujący.
Ustawienia adaptera	Głębokość kolejki transmisji/odbioru	Zalecana wartość to 150.	Wartość domyślna zazwyczaj wynosi 32.
Ustawienia TCP	Wielkości segmentów	1500 w sieci Ethernet, 4400 w sieciach Token Ring i FDDI.	Adaptery ESCON używane z protokołem TCP/IP powinny zawsze być ustawione na wartość 4096.
Ustawienia TCP	Wielkości przestrzeni transmisji/odbioru	64 kB dla obu parametrów.	Domyślną wartością jest tylko 8192 dla systemu Windows. Może być ustawiona w rejestrze systemu Windows.

Rywalizacja o zasoby systemowe

Wydajność zmniejsza się także, jeśli wiele zadań rywalizuje o zasoby systemowe. Należy uwzględnić następujące kwestie:

- Czy jednostka centralna jest całkowicie wykorzystywana? Rozważ aktualizację do nowej wersji systemu, redukcję jego obciążenia i dostrojenie systemu w celu zmniejszenia narzutu związanego z przetwarzaniem.
- Czy pamięć została całkowicie zapełniona? Rozważ aktualizację pamięci, redukcję obciążenia systemu oraz jego dostrojenie w celu zmniejszenia obciążenia pamięci.
- Czy kontroler adaptera komunikacyjnego nie jest zbyt zajęty? Rozważ zaktualizowanie sieci lub odpowiedni dobór pary kart Token Ring.
- Czy jeden z podsystemów nie jest zbyt zajęty i czy ten podsystem jest umieszczony w ścieżce danych?
- Czy w systemie nie zostały uruchomione niepotrzebne procesy lub zadania? Ogólnie przyjęto zasadę, aby nie konfigurować i nie uruchamiać usług, jeśli nie są używane regularnie, ponieważ będą one niepotrzebnie korzystać z zasobów systemowych.

- Czy kilka procesów lub zadań nie korzysta z większości zasobów? Czy można je zatrzymać? Czy można obniżyć ich priorytety? Czy można je poprawić, aby nie korzystały z tylu zasobów?

Rozwiązywanie problemów dotyczących wydajności w programie DB2 Connect

Jeśli użytkownicy programu DB2 Connect zgłaszają długi czas odpowiedzi dla dużych zapytań z serwerów bazy danych na hostach lub serwerach System i, możliwych przyczyn problemów z wydajnością należy szukać w następujących obszarach:

1. W wypadku zapytań zwracających duże bloki danych z hosta lub serwera System i (zwykle 32 kB danych i więcej) należy nadać parametrowi konfiguracyjnemu RQRIOBLK menedżera bazy danych wartość 32767. Można to zrobić, używając procesora wiersza komend w następujący sposób:

   ```
   db2 update database manager configuration using RQRIOBLK 32767
   ```

2. Należy sprawdzić, czy maksymalna wielkość RU w definicji trybu IBMRDB została ustawiona na odpowiednią wartość. Dla połączeń z wykorzystaniem sprzętu Token Ring zalecana jest wartość nie mniejsza niż 4 kB. Dla połączeń z wykorzystaniem sprzętu Ethernet czynnikiem ograniczającym może być maksymalna wielkość ramki równa 1536 bajty.

Dostrajanie programu DB2 for OS/390 and z/OS

W systemach operacyjnych OS/390 i z/OS można zoptymalizować przetwarzanie nieaktywnych wątków. W wersji 5 liczba jednocześnie przyłączonych klientów nie może przekroczyć 25 000. We wszystkich przypadkach maksymalna liczba jednocześnie aktywnych klientów wynosi 1999. Każdy klient stacji roboczej może być przyłączony również wtedy, gdy jest nieaktywny; wątek takiego klienta jest umieszczany w łańcuchu nieaktywnym przy każdym zatwierdzaniu.

Parametry DSNZPARM: CMTSTAT, CONDBAT i MAXDBAT wpływają na przetwarzanie wątków. Aby wydajność była jak największa, należy ustawić dla parametru CMTSTAT wartość INACTIVE, dopasować CONDBAT do maksymalnej liczby połączonych wątków DBAT zapewniającej wysoką wydajność i ustawić parametr MAXDBAT na największą akceptowalną liczbę aktywnych wątków DBAT.

Zwiększanie szybkości przesyłania danych przez oprogramowanie DB2 Connect

Poza łączeniem w bloki wierszy zestawu wynikowego zapytania produkt DB2 for OS/390 and z/OS może również w odpowiedzi na żądanie OPEN lub FETCH zwracać do zdalnego klienta, takiego jak DB2 Connect, wiele takich bloków zapytania. Zamiast cyklicznego wysyłania przez klienta do serwera DB2 for OS/390 and z/OS żądań pojedynczych bloków danych wierszy za każdym razem, klient może opcjonalnie zażądać, aby serwer zwracał pewną liczbę bloków zapytania dodatkowo poza tym, który zawsze jest zwracany. Bloki te są nazywane dodatkowymi blokami zapytania.

Ta nowa opcja pozwala klientowi ograniczyć odwracanie linii sieciowej, które stanowi główny składnik kosztów wpływających na wydajność sieci. Ograniczenie liczby żądań bloków zapytania wysyłanych przez klienta do serwera przekłada się na znaczny wzrost wydajności. Wynika on z faktu, że przełączanie między trybem nadawania i odbioru jest operacją kosztowną z punktu widzenia wydajności. Produkt DB2 Connect może korzystać teraz z tej możliwości zwiększenia wydajności przez domyślne żądanie dodatkowych bloków zapytania od serwera DB2 for OS/390 and z/OS.

Aby dla preferowanego protokołu sieciowego TCP/IP w pełni korzystać z możliwości zwracania dodatkowych bloków zapytania (każdy może mieć wielkość do 32 kB), w produkcie DB2 Connect włączone zostały zgodne ze specyfikacją RFC-1323 rozszerzenia skalowania okna. Ta opcja umożliwia dynamiczne korygowanie wielkości okien wysyłania i odbierania w protokole TCP/IP w celu efektywnego dostosowania ich do potencjalnie olbrzymich ilości danych zwracanych w dodatkowych blokach zapytania.

Dodatkowy blok zapytania

Obsługa dodatkowego bloku zapytania na serwerach z bazą danych DB2 Universal Database (UDB) for OS/390 and z/OS w wersji 7 lub późniejszej jest skonfigurowana poprzez parametr EXTRA BLOCKS SRV na panelu instalacyjnym narzędzia DB2 DDF. Konfigurowanie tej obsługi polega na sterowaniu maksymalną liczbą dodatkowych bloków zapytania, które program DB2 może odesłać do klienta w odpowiedzi na żądanie. Parametrowi temu można nadać wartość z zakresu od 0 do 100. Nadanie parametrowi wartości 0 powoduje wyłączenie wysyłania dodatkowych bloków zapytania. Aby najpełniej wykorzystać tę opcję, należy zawsze używać wartości domyślnej, która wynosi 100, eliminując przy tym wszelkie przeszkody w sieci, które nakazywałyby obniżenie tej wartości.

Po stronie klienta, gdzie aplikacja uzyskuje dostęp do produktu DB2 for z/OS bezpośrednio poprzez zainstalowany w tym samym miejscu produkt DB2 Connect lub poprzez zainstalowany na innym serwerze produkt DB2 Connect, istnieją różne sposoby aktywowania obsługi odpowiedniego produktu DB2 Connect na podstawie kursora lub instrukcji:

- wykorzystanie wielkości zestawu wierszy zapytania dla kursora
- użycie klauzuli 'OPTIMIZE for N ROWS' w instrukcji SELECT skojarzonej z kursorem
- użycie klauzuli 'FETCH FIRST N ROWS' w instrukcji SELECT skojarzonej z kursorem

Program DB2 Connect może włączyć obsługę dodatkowych bloków zapytania przy użyciu różnych funkcji API SQL:

Wbudowany SQL

- Użytkownik może wywołać dla zapytania obsługę dodatkowego bloku zapytania przez określenie klauzuli OPTIMIZE for N ROWS, FETCH FIRST N ROWS ONLY lub obu wewnątrz instrukcji SELECT.
- Podanie klauzuli OPTIMIZE for N ROWS oznacza, że produkt DB2 for OS/390 and z/OS będzie próbował połączyć w blok wymaganą liczbę wierszy zwracanych do produktu DB2 Connect, zgodnie z ustawieniem parametru instalacyjnego EXTRA BLOCKS SRV DDF. Aplikacja może wybrać pobieranie większej niż N liczby wierszy, ponieważ produkt DB2 for z/OS nie ogranicza do N całkowitej liczby wierszy, które mogą zostać ostatecznie zwrócone jako zestaw wynikowy zapytania.
- Klauzula FETCH FIRST N ROWS ONLY działa w podobny sposób, z wyjątkiem tego, że zestaw wynikowy zapytania jest ograniczony do N wierszy przez produkt DB2 for OS/390 and z/OS. Pobranie większej liczby wierszy niż N spowodowałoby wygenerowanie kodu SQL +100 (koniec danych).

CLI/ODBC

- Użytkownik może wywołać obsługę dodatkowych bloków zapytania, korzystając z atrybutu instrukcji SQL_MAX_ROWS.
- Klauzula FETCH FIRST N ROWS ONLY używana jest natomiast w przypadku serwera DB2 UDB for OS/390 and z/OS w wersji 7.1 lub późniejszej.
  - W wersji 7 zestaw wynikowy zapytania jest ograniczony do N wierszy przez produkt DB2 for OS/390 and z/OS. Próba pobrania większej liczby wierszy niż N spowodowałaby zwrócenie kodu SQL_NO_DATA_FOUND.

– W wersji 8 i późniejszych interfejs CLI gwarantuje, że tylko pierwszych N wierszy jest zwracanych do aplikacji poprzez menedżer kursorów klienta.

JDBC Użytkownik może wywołać obsługę dodatkowych bloków zapytania, korzystając z metody setMaxRows. Podobnie jak przy włączaniu interfejsu CLI/ODBC, produkt DB2 Connect zaznaczy klauzulę OPTIMIZE for N ROWS w przypadku serwera DB2 for OS/390 and z/OS w wersji 6.x. Produkt DB2 Connect zaznaczy również klauzulę FETCH FIRST N ROWS ONLY w przypadku serwera DB2 for z/OS w wersji 7.1 lub późniejszej.

Skalowanie okien RFC-1323

Skalowanie okien jest obsługiwane na wszystkich platformach Windows, Linux i UNIX, które obsługują zgodne ze specyfikacją RFC-1323 rozszerzenia protokołu TCP/IP. Tę opcję można włączyć w produkcie DB2 for Windows, Linux, or UNIX, korzystając ze zmiennej rejestrowej DB2SORCVBUF produktu DB2. Aby włączyć skalowanie okien, należy nadać tej zmiennej rejestru wartość większą niż 64 kB. Na przykład w produkcie DB2 for Windows, Linux, or UNIX można wydać komendę db2set DB2SORCVBUF =65537.

Maksymalne wielkości buforów wysyłania i odbierania zależą od określonego systemu operacyjnego. Aby sprawdzić, czy skonfigurowane wielkości buforów zostały zaakceptowane, użytkownik może nadać parametrowi konfiguracyjnemu menedżera bazy danych DIAGLEVEL wartość 4 (poziom informacyjny) i przejrzeć komunikaty w dzienniku powiadomień administratora.

Aby skalowanie okien przyniosło efekt, musi być włączone po obu stronach połączenia; zarówno na stacji roboczej, jak i na hoście, bezpośrednio przez stos TCP/IP systemu operacyjnego, bądź pośrednio poprzez produkt DB2. Na przykład w przypadku produktu DB2 for z/OS skalowanie okien może obecnie być aktywowane tylko poprzez system operacyjny przez ustawienie parametru TCPRCVBUFRSIZE na dowolną wartość wyższą niż 64K. Jeśli zdalny klient IBM Data Server Client uzyskuje dostęp do hosta lub bazy danych System i DB2 za pośrednictwem stacji roboczej serwera DB2 Connect, można również włączyć skalowanie okien po stronie klienta. Korzystając z tego samego elementu, można również włączyć skalowanie okien między zdalnym klientem IBM Data Server Client a stacją roboczą serwera DB2, jeśli nie bierze w tym udziału host ani baza danych System i DB2.

Chociaż skalowanie okien ma na celu zwiększenie wydajności sieci, należy podkreślić, że cel ten nie zawsze zostaje zrealizowany. Kombinacje takich czynników, jak wielkość ramki używana przez adapter sieci Ethernet lub Token Ring, wielkość IP MTU i inne ustawienia routerów w trakcie połączenia, mogą spowodować nawet obniżenie wydajności przy włączonym skalowaniu okien. Dlatego domyślnie skalowanie okien jest wyłączone, a bufory wysyłania i odbierania mają wielkość 64 kB.

Należy ocenić skutki włączenia funkcji skalowania okien i wykonać wszelkie niezbędne korekty ustawień sieci. Podstawowe informacje dotyczące strojenia sieci w celu poprawienia jej wydajności można znaleźć pod adresem http://www.networking.ibm.com/nhd/webnav.nsf/pages/netdocs.html.

Konwersja danych hosta

Podczas przekazywania danych pomiędzy różnymi środowiskami (takimi jak Intel [Windows], IEEE [systemy operacyjne Linux i UNIX], zSeries [VM, VSE, z/OS], System i [OS/400]), liczbowe typy danych (takie jak dziesiętne, całkowite, zmiennoprzecinkowe) mogą wymagać konwersji. Konwersja ta może mieć wpływ na wydajność.

Koszt jednostki centralnej związany z konwersją znaków jednobajtowych jest generalnie mniejszy niż koszt konwersji danych numerycznych (gdzie konwersja danych jest wymagana).

Koszt konwersji danych DATE/TIME/TIMESTAMP jest prawie taki sam, jak koszt konwersji danych jednobajtowych CHAR. Najdroższa jest konwersja danych zmiennopozycyjnych FLOATING. Projektant aplikacji może skorzystać z tych faktów, projektując aplikację w oparciu o produkt DB2 Connect.

Jeśli tabela bazy danych ma kolumnę zdefiniowaną 'FOR BIT DATA', nie jest wymagane wykonywanie konwersji danych znakowych przekazywanych między aplikacją i bazą danych. Kolumny tej można użyć podczas archiwizacji danych na serwerze bazy danych na hoście lub serwerze System i.

Typy danych znakowych

Dane znakowe mogą być typu CHAR lub VARCHAR. Wydajność obsługi danych zależy od typowej długości danych w polu:

- Jeśli wielkość danych rzeczywistych zmienia się znacząco, wydajniejszy jest typ VARCHAR, ponieważ CHAR dodaje dodatkowe puste znaki, aby zapełnić pole. Te puste znaki muszą być przesyłane przez sieć tak samo, jak inne znaki.
- Jeśli wielkość danych rzeczywistych nie zmienia się znacząco, bardziej wydajny jest typ CHAR, ponieważ każde pole VARCHAR ma kilka bajtów informacji na temat długości, które też muszą zostać przesłane.

Sprzęt sieciowy

Poniższe informacje dotyczą sprzętu:

- Prędkość sieci lub nośnika transmisji

  Szybszy nośnik transmisji poprawia wydajność. Poniżej podano przykładowe prędkości przesyłania danych:

  Kanał-kanał (technika światłowodowa)
  4,0 MB/s

  Sieć LAN o szybkości 16 MB/s
  2,0 MB/s

  Kanał-kanał (zwykłe)
  1,0 MB/s

  Sieć LAN o szybkości 4 MB/s
  0,5 MB/s

  Nośna T1 o dużej szybkości (1,544 MB/s)
  0,193 MB/s

  Szybka zdalna linia telefoniczna 56 kB/s
  0,007 MB/s

  Modem 19,6 kB/s
  0,002 MB/s

  Modem 9600 b/s
  0,001 MB/s.

  Szybkość przesyłania danych jest ograniczona przez najwolniejszy nośnik transmisji na drodze do serwera bazy danych na hoście lub serwerze System i.

- Adapter sieciowy lub sterownik komunikacyjny

Należy uważnie planować użytkowanie pamięci adaptera sieciowego oraz sterownika komunikacyjnego. Dodatkowo należy razem ze specjalistą od sieci sprawdzić, czy kontroler może obsługiwać dodatkowy ruch wygenerowany przez program DB2 Connect.

- Topologia sieci

  Jeśli dane są przekazywane pomiędzy sieciami LAN lub innymi sieciami, należy wziąć pod uwagę potrzebny na to czas. Na całkowity czas składają się również przejścia przez mosty, routery i gatewaye. Na przykład zmniejszenie liczby mostów, przez które przechodzą dane, zredukuje liczbę przeskoków wymaganych dla każdego żądania.

  Należy również rozważyć fizyczną odległość między węzłami. Nawet jeśli komunikat jest przesyłany przez satelitę, czas transferu jest ograniczony przez prędkość światła (3 * 10**8 m/s) oraz odległość transmisji tam i z powrotem między nadawcą i odbiorcą.

- Ruch w sieci

  Jeśli przepustowość sieci zostanie w pełni wykorzystana, zmniejszy się zarówno szerokość pasma, jak i prędkość przesyłania danych dla pojedynczej aplikacji.

  Gdy dane zostaną zgromadzone, w pewnym miejscu sieci może wystąpić przeciążenie; na przykład w starym NCP z buforem bardzo małej wielkości.

- Niezawodność sieci

  Jeśli błędy w sieci występują często, przepustowość sieci zmniejsza się i powoduje to złą wydajność z powodu konieczności ponownej transmisji danych.

Rozdział 20. Dostrajanie wydajności aplikacji CLI/ODBC

CLI/ODBC jest aplikacyjnym interfejsem programowym języka SQL, który może być wywoływany przez aplikacje bazy danych. Funkcje CLI wywołują procedury składowane DB2, które z kolei uzyskują dostęp do tabel katalogów systemowych.

Niektóre aplikacje korzystają z interfejsów API ODBC do zbierania informacji (metadanych) używanych w trakcie dalszego przetwarzania. Dziesięć dopuszczalnych wywołań API dotyczących metadanych to:

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns

Niektóre spośród aplikacji CLI/ODBC, które używają wymienionych powyżej funkcji API metadanych mogą wykonywać zapytania dla wszystkich obiektów w bazie danych. Na przykład wywołanie SQLTables żąda metadanych dla wszystkich tabel w bazie danych. W przypadku dużego systemu takie żądania mogą powodować duży ruch w sieci, zabierać dużo czasu i zużywać znaczną ilość zasobów serwera.

Niektóre spośród parametrów inicjalizacyjnych CLI/ODBC mogą być użyte do ograniczenia ilości danych zwracanych przez początkowe wywołania funkcji API na etapie "zbierania informacji" po pierwszym uzyskaniu połączenia z bazą danych. Parametry te można ustawić korzystając z następujących metod:

1. Ręczna edycja pliku db2cli.ini.
2. Zmiana ustawień ODBC/CLI dla bazy danych przy użyciu Asysty podczas konfigurowania klienta (na platformach ją obsługujących).
3. Aktualizacja konfiguracji CLI bazy danych przy użyciu interfejsu poziomu wywołania (CLI) administratora DBA.

Do parametrów należą:

- DBName
- TableType
- SchemaList
- SysSchema
- GrantorList
- GranteeList

© Copyright IBM Corp. 1993, 2010

Część 5. Rozwiązywanie problemów

© Copyright IBM Corp. 1993, 2010

Rozdział 21. Rozwiązywanie problemów

W skład środowiska DB2 Connect wchodzi wiele produktów, takich jak oprogramowanie, sprzęt i produkty do obsługi komunikacji. Najlepszym podejściem do rozwiązywania problemów jest metoda eliminacji i uściślania dostępnych danych, tak aby dojść do prawidłowego wniosku (znaleźć miejsce wystąpienia błędu).

Po wybraniu odpowiedniego tematu i po zgromadzeniu niezbędnych informacji należy przejść do odpowiedniej części.

Zbieranie odpowiednich informacji

Rozwiązywanie problemów obejmuje zawężanie obszaru wystąpienia problemu i zbadanie jego przyczyn. Właściwym punktem początkowym jest zgromadzenie odpowiednich informacji i określenie: co wiadomo, jakie dane nie zostały zebrane oraz jakie ścieżki można wyeliminować. Należy odpowiedzieć na następujące pytania:

- Czy ustanowienie połączenia początkowego przebiegało pomyślnie?
- Czy sprzęt działa prawidłowo?
- Czy połączenia komunikacyjne działają?
- Czy nastąpiły zmiany w sieci komunikacyjnej, które mogły spowodować, że poprzednie pozycje katalogu stały się niepoprawne?
- Czy baza danych została uruchomiona?
- Czy występuje przerwa w komunikacji między jednym lub większą liczbą klientów a serwerem DB2 Connect (bramą); między bramą DB2 Connect a serwerem bazy danych na hoście lub serwerze System i czy też między programem DB2 Connect Personal Edition a serwerem bazy danych na hoście lub serwerze System i?
- Co można wywnioskować na podstawie zawartości komunikatu i znaczników zwróconych w komunikacie?
- Czy użycie narzędzi diagnostycznych, takich jak db2trc, db2pd lub db2support, może okazać się pomocne?
- Czy inne maszyny wykonujące podobne zadania działają poprawnie?
- Czy, jeśli jest to zadanie zdalne, jest ono lokalnie wykonywane pomyślnie?

Połączenie początkowe nie powiodło się

Przejrzyj następujące pytania i sprawdź, czy wykonano niezbędne czynności instalacyjne:

1. *Czy proces instalacji zakończył się pomyślnie?*
   - Czy całe wstępnie wymagane oprogramowanie było dostępne?
   - Czy spełnione były wymagania dotyczące pamięci i miejsca na dysku?
   - Czy zainstalowano obsługę klienta zdalnego?
   - Czy instalacja oprogramowania komunikacyjnego zakończyła się bez wystąpienia błędów?
2. *Czy w przypadku systemów operacyjnych UNIX utworzona została instancja produktu?*
   - Czy administrator utworzył użytkownika i grupę, która ma być właścicielem instancji i grupą sysadm?
3. *Jeśli mają zastosowanie informacje o licencji, to czy zostały ona pomyślnie przetworzone?*
   - Czy w przypadku systemów operacyjnych UNIX w pliku nodelock wprowadzono hasło dostarczone przez firmę IBM?

© Copyright IBM Corp. 1993, 2010

4. *Czy poprawnie skonfigurowano komunikację między serwerem bazy danych na hoście lub serwerze System i a stacją roboczą?*
   - Należy rozważyć trzy konfiguracje:
     a. Konfiguracja serwera bazy danych na hoście lub serwerze System i identyfikuje na serwerze requester aplikacji. W systemie zarządzania baz danych na hoście lub serwerze System i dostępne są pozycje katalogu definiujące requester pod względem położenia, protokołu sieciowego i zabezpieczeń.
     b. Konfiguracja stacji roboczej DB2 Connect definiuje populację klientów dla serwera oraz hosta lub serwer System i dla klienta.
     c. Konfiguracja stacji roboczej klienta musi mieć zdefiniowaną nazwę stacji roboczej i protokołu komunikacyjnego.
   - Analiza problemu nie nawiązania początkowego połączenia obejmuje sprawdzenie poprawności i kompletności nazw jednostek fizycznych lub sprawdzenie prawidłowości podanego numeru portu i nazwy hosta dla połączeń TCP/IP.
   - Zarówno dla administratora bazy danych na hoście lub serwerze System i, jak i dla administratorów sieci dostępne są narzędzia umożliwiające diagnozowanie problemów
5. *Czy dysponujesz wymaganym przez hosta lub system zarządzania bazą danych serwera System i poziomem uprawnień do korzystania z hosta lub bazy danych serwera System i?*
   - Przeanalizuj uprawnienia dostępu użytkownika, reguły dla kwalifikatorów tabel, przewidywane wyniki.
6. *Czy próba zastosowania procesora wiersza komend (CLP) do wysłania instrukcji SQL dla hosta lub serwera bazy danych System i kończy się niepowodzeniem?*
   - Czy wykonano procedurę powiązania procesora wiersza komend z hostem lub serwerem bazy danych System i?

Problemy występujące po połączeniu początkowym

Aby zawęzić liczbę możliwych przyczyn wystąpienia problemu, należy odpowiedzieć na następujące pytania:

1. *Czy występują szczególne lub niecodzienne okoliczności związane z działaniem?*
   - Czy jest to nowa aplikacja?
   - Czy są używane nowe procedury?
   - Czy ostatnio nastąpiły zmiany, które mogły wpłynąć na system? Czy zmieniono jakieś oprogramowanie lub aplikacje od ostatniego pomyślnego uruchomienia aplikacji lub scenariusza?
   - W przypadku aplikacji: jakiego interfejsu API użyto do utworzenia programu?
   - Czy w systemie użytkownika uruchomiono inne aplikacje korzystające z oprogramowania lub komunikacyjnych interfejsów API?
   - Czy zainstalowano ostatnio pakiet poprawek? Jeśli problem wystąpił, gdy użytkownik próbował użyć opcji, która nie była używana (lub załadowana) w systemie operacyjnym od czasu zainstalowania, należy określić najnowszy poziom pakietu poprawek i zainstalować ten pakiet *po* zainstalowaniu danej opcji.
2. *Czy ten błąd wystąpił już wcześniej?*
   - Czy istnieją udokumentowane rozwiązania poprzednich błędów?
   - Kto zauważył poprzedni błąd i czy te osoby mogą pomóc w ustaleniu ewentualnych działań, które można podjąć?
3. *Czy zbadano problem przy użyciu komend oprogramowania komunikacyjnego zwracających informacje o sieci?*
   - Protokół TCP/IP może dysponować cenną informacją uzyskaną przez komendy i demony TCP/IP.

4. *Czy informacje umieszczone w obszarze komunikacyjnym SQL są pomocne?*
   - Procedury obsługi problemu powinny zawierać kroki, w których sprawdzana jest zawartość pól SQLCODE i SQLSTATE.
   - Kody SQLSTATE umożliwiają programistom aplikacji testowanie pod kątem występowania klas błędów wspólnych dla rodziny produktów baz danych DB2. To pole może stanowić wspólną bazę w sieci rozproszonej relacyjnej bazy danych.
5. *Czy na serwerze wykonano komendę DB2START?* Ponadto należy sprawdzić, czy poprawnie ustawiono zmienną środowiskową DB2COMM dla klientów uzyskujących zdalny dostęp do serwera.
6. *Czy inne komputery wykonujące to samo zadanie mogą pomyślnie połączyć się z serwerem?* Możliwe, że została osiągnięta maksymalna liczba klientów próbujących nawiązać połączenie z serwerem. Jeśli inny klient odłączy się od serwera, to czy klient, który poprzednio nie mógł się połączyć, może to obecnie wykonać?
7. *Czy komputer ma poprawny sposób adresowania?* Sprawdź, czy maszyna jest unikalna w sieci.
8. *Czy klientowi nadano odpowiednie uprawnienia, jeśli łączy się on zdalnie?* Nawiązanie połączenia z instancją mogło się powieść, ale uwierzytelnienie na poziomie bazy danych lub tabeli mogło zakończyć się niepowodzeniem.
9. *Czy jest to pierwszy komputer, który łączy się ze zdalną bazą danych?* W rozproszonych środowiskach routery lub mosty międzysieciowe mogą blokować komunikację pomiędzy klientem a serwerem. Na przykład używając protokołu TCP/IP upewnij się, że w odniesieniu do zdalnego hosta działa komenda PING.

Nieobsługiwane komendy DDM

Komendy DDM BNDCPY, BNDDPLY, DRPPKG i DSCRDBTBL nie są obsługiwane w programie DB2 wersja 9.5 dla systemów Linux, UNIX i Windows, który działa jako serwer aplikacji DRDA (DRDA AS).

Objawy

Jeśli requester aplikacji DRDA (DRDA AR) łączy się z bazą danych DB2 9.5 dla systemów Linux, UNIX i Windows i wydaje następujące komendy, zakończą się one niepowodzeniem:

Tabela 18. Nieobsługiwane komendy DDM

Komenda DDM	Punkt kodowy DDM	Opis
BNDCPY	X'2011'	Kopiuj istniejący pakiet relacyjnej bazy danych (RDB)
BNDDPLY	X'2016'	Instaluj istniejący pakiet RDB
DRPPKG	X'2007'	Usuń pakiet
DSCRDBTBL	X'2012'	Opisz tabelę RDB

Dodatkowo, następujące punkty kodowe, używane w deskryptorach SQLDTA, również nie są obsługiwane:

Tabela 19. Nieobsługiwane obiekty danych FD:OCA

Obiekty danych FD:OCA	Punkt kodowy DDM	Opis
FDOEXT	X'147B'	Przydziały danych FD:OCA
FDOOFF	X'147D'	Przesunięcia danych FD:OCA

Najczęstszym błędem w tej sytuacji jest SQL30020N ("Uruchomienie nie powiodło się z powodu błędu protokołu dystrybucyjnego, który będzie miał wpływ na poprawne

uruchomienie kolejnych komend lub instrukcji SQL").

Przyczyny

Architektura DDM jest częścią protokołu DRDA. Komendy DDM BNDCPY, BNDDPLY, DRPPKG i DSCRDBTBL istnieją we wszystkich wersjach protokołu DRDA obsługiwanych przez program DB2 wersja 9.5 dla systemów Linux, UNIX i Windows, ale serwer aplikacji DRDA nie obsługuje tych komend DDM.

Dodatkowo serwer ten nie obsługuje punktów kodowych **FDOEXT** i **FDOOFF**. Te punkty kodowe są używane w deskryptorze SQLDTA wysyłanym do serwera po otrzymaniu żądania operacji wejściowej dla tablicy kolumnowej.

Diagnozowanie problemu

W odpowiedzi na nieobsługiwane komendy w pliku śledzenia DB2 dla serwera aplikacji DRDA znajdzie się komunikat podobny do poniższego: ERROR MSG = Parser: Command Not Supported.

Rozwiązanie problemu

Obecnie nie ma żadnych komend alternatywnych w stosunku do komend DDM BNDCPY i BNDDPLY.

Aby usunąć pakiet, użyj instrukcji SQL DROP PACKAGE. Na przykład, połącz się z serwerem aplikacji DRDA DB2 wersja 9.5 dla systemów Linux, UNIX i Windows i wyślij instrukcję DROP PACKAGE w żądaniu EXECUTE IMMEDIATE. Program DB2 wersja 9.5 dla systemów Linux, UNIX i Windows natychmiast rozpocznie przetwarzanie żądania.

Aby opisać tabelę RDB, należy użyć jednej z następujących komend DDM: DSCSQLSTT (opisz instrukcję SQL) lub PRPSQLSTT (przygotuj instrukcję SQL). Na przykład, jeśli chcesz uzyskać opis tabeli TAB1, opisz lub przygotuj następującą instrukcję: SELECT * FROM TAB1.

Uwaga: Kiedy requester aplikacji DRDA wydaje komendę PRPSQLSTT, konieczne jest również określenie zmiennej instancji **RTNSQLDA** z wartością TRUE, w przeciwnym wypadku deskryptor SQLDARD nie zostanie zwrócony przez serwer.

Aby uniknąć problemów dotyczących punktów kodowych **FDOEXT** i **FDOOFF** , należy używać żądań operacji wejściowej dla tablicy wierszowej zamiast żądań operacji wejściowej dla tablicy parametrowej (lub kolumnowej).

Narzędzia diagnostyczne

Jeśli wystąpi problem, można użyć następujących narzędzi diagnostycznych:

- Wszystkie informacje diagnostyczne, w tym pliki zrzutu, pliki pułapek, dzienniki błędów, pliki powiadomień i dzienniki alertów, znajdują się w ścieżce określonej przez parametr konfiguracyjny menedżera bazy danych definiujący ścieżkę katalogu zawierającego dane diagnostyczne (**diagpath**):

  Jeśli ten parametr konfiguracyjny ma wartość NULL, dane diagnostyczne są zapisywane w jednym z następujących katalogów lub folderów:

  – W środowiskach Linux i UNIX: INSTHOME/sqllib/db2dump, gdzie *INSTHOME* to katalog osobisty instancji.
  – W przypadku obsługiwanych środowisk Windows:

- Jeśli zmienna środowiskowa **DB2INSTPROF** nie jest ustawiona, używana jest ścieżka x:\SQLLIB\DB2INSTANCE, gdzie x:\SQLLIB to dysk i katalog określony w zmiennej rejestrowej **DB2PATH**, a wartością parametru **DB2INSTANCE** jest nazwa instancji.

  Uwaga: Nazwa katalogu może być inna niż SQLLIB.
- Jeśli zmienna środowiskowa **DB2INSTPROF** jest ustawiona, używana jest ścieżka x:\DB2INSTPROF\DB2INSTANCE, gdzie **DB2INSTPROF** to nazwa katalogu profilu instancji, a **DB2INSTANCE** to nazwa instancji (domyślnie wartość **DB2INSTDEF** w 32-bitowych systemach operacyjnych Windows).

- W przypadku systemów operacyjnych Windows dziennik powiadomień administracyjnych można przejrzeć, korzystając z przeglądarki zdarzeń.
- Dostępne narzędzia diagnostyczne to między innymi **db2trc**, **db2pd** i **db2support**.
- W przypadku systemów operacyjnych Linux i UNIX komenda **ps**, która na standardowym wyjściu zwraca informacje o statusie aktywnych procesów.
- W przypadku systemów operacyjnych UNIX podstawowy plik tworzony w bieżącym katalogu w razie wystąpienia poważnych błędów. Zawiera on obraz pamięci przerwanego procesu i może być użyty do określenia funkcji, która spowodowała błąd.

Rozdział 22. Śledzenie DB2 w programie DB2 Connect

Zapisy śledzenia dostarczają ważnych informacji pomocnych podczas rozwiązywania problemów. Możliwe jest uzyskanie, zrzut lub sformatowanie śledzenia z produktu serwerowego DB2. Narzędzie śledzenia dostarczane jest jako część produktu serwerowego DB2.

Uzyskiwanie pliku śledzenia DB2 przy użyciu narzędzia db2trc

Komenda **db2trc** steruje narzędziem do śledzenia udostępnianym przez produkt DB2. Narzędzie do śledzenia rejestruje informacje o operacjach oraz formatuje je do czytelnej postaci.

Należy być świadomym, że uruchomione śledzenie wymaga dodatkowego nakładu pracy, tak więc włączenie narzędzia śledzenia może wpłynąć na wydajność systemu.

Generalnie z funkcji śledzenia produktu DB2 korzystają zespoły programistyczne oraz dział wsparcia DB2 przy rozwiązywaniu problemów. Użytkownik może uruchomić śledzenie, aby uzyskać informacje na temat problemu, który sprawdza, ale ich przydatność bez znajomości kodu źródłowego produktu DB2 jest raczej ograniczona.

Pomimo tego, wiedza o poprawnym włączaniu śledzenia i zrzucaniu plików śledzenia jest ważna, w razie konieczności ich uzyskania.

Uwaga: Aby korzystać z komendy db2trc, wymagane są uprawnienia SYSADM, SYSCTRL lub SYSMAINT.

Aby poznać ogólne przeznaczenie dostępnych opcji, komendę db2trc należy wywołać bez parametrów:

```
C:\>db2trc
Składnia: db2trc (chg|clr|dmp|flw|fmt|inf|off|on) opcje
```

Więcej informacji na temat określonego parametru komendy db2trc można uzyskać korzystając z opcji -u. Na przykład aby sprawdzić informacje na temat włączania śledzenia, należy wykonać następującą komendę:

```
db2trc on -u
```

Wyświetlone zostaną informacje o wszystkich opcjach dodatkowych (oznaczone jako "facilities"), które można podać podczas włączania śledzenia DB2.

Najważniejszą opcją podczas włączania śledzenia jest opcja -L. Określa wielkość buforu pamięci, który zostanie wykorzystany do przechowywania zapisu śledzonych informacji. Wielkość buforu można podać w bajtach lub megabajtach. Aby podać ją w megabajtach, do wartości należy dodać literę "M" lub "m". Wielkość buforu pamięci musi być potęgą dwóch megabajtów. Jeśli podana zostanie wielkość, która nie spełnia tego wymagania, wielkość buforu zostanie automatycznie zaokrąglona w dół do najbliższej potęgi dwójki.

Jeśli bufor jest zbyt mały, informacje mogą zostać utracone. Domyślnie, jeśli bufor zapełni się całkowicie, przechowywane są tylko najnowsze informacje śledzenia. Jeśli bufor jest zbyt duży, jego wysłanie do działu wsparcia DB2 może być utrudnione.

W przypadku śledzenia operacji, która jest relatywnie krótka (takiej jak połączenie z bazą danych), zazwyczaj wystarczająca jest wielkość około 8 MB:

© Copyright IBM Corp. 1993, 2010

```
C:\> db2trc on -l 8M
Śledzenie jest włączone
```

Jednak większy bufor śledzenia może być wymagany w przypadku śledzenia większej operacji lub wykonywania wielu operacji w tym samym czasie.

Na większości platform śledzenie można włączyć w dowolnym momencie i będzie ono działać, tak jak opisano powyżej. Jednak istnieje kilka sytuacji, których należy być świadomym:

1. W systemach partycjonowanej bazy danych z wieloma partycjami śledzenie trzeba uruchamiać dla każdej fizycznej (w przeciwieństwie do logicznej) partycji bazy danych.
2. Na platformach HP-UX, Linux i Solaris jeśli śledzenie zostanie wyłączone po uruchomieniu instancji, przy następnym uruchamianiu śledzenia, niezależnie od podanej wielkości, zostanie zastosowany bardzo mały bufor. Na przykład wczoraj śledzenie zostało uruchomione za pomocą komendy db2trc on -l 8m, następnie zebrano informacje i wyłączono śledzenie (db2trc off). Dzisiaj użytkownik chce uruchomić śledzenie z użyciem buforu pamięci o wielkości 32 megabajtów (db2trc on -l 32m) bez wyłączania instancji i jej restartowania. Okaże się, że w tym przypadku śledzenie będzie miało dostępny jedynie mały bufor. Aby skutecznie uruchomić śledzenie na tych platformach, przed uruchomieniem instancji należy je włączyć, podając wymaganą wielkość buforu, a następnie należy "wyczyścić" bufor (jeśli jest to konieczne).

Zrzut pliku śledzenia DB2

Po włączeniu narzędzia śledzenia za pomocą opcji on śledzone będą wszystkie kolejne operacje przeprowadzane przez instancję.

Podczas działania funkcji śledzenia można wyczyścić bufor śledzenia za pomocą opcji clr. Spowoduje to usunięcie wszystkich informacji zawartych w buforze śledzenia.

```
C:\>db2trc clr
Śledzenie zostało wyczyszczone
```

Po zakończeniu działania śledzonej operacji należy użyć opcji dmp, po której należy podać nazwę pliku dla zrzutu buforu pamięci na dysk. Na przykład:

```
C:\>db2trc dmp trace.dmp
Śledzenie zostało zrzucone do pliku
```

Po zrzuceniu buforu śledzenia na dysk narzędzie śledzenia będzie kontynuowało swoje działanie. Aby wyłączyć śledzenie, należy użyć opcji off:

```
C:\>db2trc off
Śledzenie jest wyłączone
```

Formatowanie pliku śledzenia DB2

Plik zrzutu utworzony przez komendę db2trc dmp ma format binarny i jest nieczytelny.

Aby sprawdzić, czy plik śledzenia może być odczytany, należy sformatować jego postać binarną, aby przedstawić sterowanie przepływem i wysłać sformatowane dane wyjściowe do urządzenia null. Poniższy przykład prezentuje komendę umożliwiającą wykonanie tego zadania:

```
db2trc flw przyklad.trc nul
```

gdzie przyklad.trc jest plikiem binarnym, który został utworzony za pomocą opcji dmp.

Dane wyjściowe tej komendy jawnie wskażą, czy podczas odczytu pliku wystąpił problem oraz czy śledzenie zostało zawinięte.

W tym momencie plik zrzutu można wysłać do działu wsparcia DB2. Pracownicy działu wsparcia powinni następnie sformatować go w oparciu o poziom usług użytkownika DB2. Jednak czasami mogą poprosić o sformatowanie pliku zrzutu do formatu ASCII przed jego wysłaniem. Taką operację można przeprowadzić za pomocą opcji flw i fmt. Uruchamiając komendę, należy podać nazwę binarnego pliku zrzutu razem z nazwą pliku ASCII, który ma zostać utworzony:

```
C:\>db2trc flw trace.dmp trace.flw
C:\Temp>db2trc flw trace.dmp trace.flw
Total number of trace records     : 18854
Trace truncated                   : NO
Trace wrapped                     : NO
Number of trace records formatted : 1513 (pid: 2196 tid 2148 node: -1)
Number of trace records formatted : 100 (pid: 1568 tid 1304 node: 0)
...

C:\>db2trc fmt trace.dmp trace.fmt
C:\Temp>db2trc fmt trace.dmp trace.fmt
Trace truncated                   : NO
Trace wrapped                     : NO
Total number of trace records     : 18854
Number of trace records formatted : 18854
```

Jeśli w pozycji "Trace wrapped" (śledzenie zawinięte) znajduje się wartość "YES" (tak), oznacza to, że bufor śledzenia nie był wystarczająco duży, aby pomieścić wszystkie informacje zebrane w trakcie śledzenia. W zależności od sytuacji, zawinięte informacje śledzenia mogą być poprawne. Jeśli użytkownik jest zainteresowany najnowszymi informacjami (to znaczy informacjami zachowywanymi domyślnie, chyba że podano opcję -i), wtedy zawartość pliku śledzenia może być wystarczająca. Jeśli jednak jest zainteresowany tym co zdarzyło się na początku okresu śledzenia lub wszystkim co wystąpiło, konieczne może być powtórzenie operacji z użyciem większego buforu śledzenia.

Podczas formatowania pliku binarnego do postaci możliwego do odczytania pliku tekstowego dostępnych jest kilka opcji. Na przykład do przekonwertowania danych binarnych i umieszczenia wyniku w formacie xml możliwym do analizowania można użyć komendy db2trc fmt -xml trace.dmp trace.fmt. Opcje dodatkowe są przedstawione w szczegółowym opisie komendy śledzenia (db2trc).

Inną rzeczą, o której należy wiedzieć, jest to, że w systemach operacyjnych Linux i UNIX produkt DB2 dokona automatycznego zrzutu buforu śledzenia na dysk w przypadku wyłączenia instancji z powodu poważnego błędu. A zatem, gdy instancja jest wyłączana nieprawidłowo i jest włączone śledzenie, w katalogu diagnostycznym zostanie utworzony plik o nazwie db2trdmp.###, gdzie ### jest numerem partycji bazy danych. Nie dzieje się tak na platformach Windows. W takich sytuacjach dane śledzenia należy zrzucić ręcznie.

Podsumowując, poniżej znajduje się przykład typowej sekwencji komend db2trc:

```
db2trc on -l 8M
db2trc clr
<Wykonaj komendy odtwarzające problem>
db2trc dump db2trc.dmp
db2trc off
db2trc flw db2trc.dmp <nazwa_pliku>.flw
db2trc fmt db2trc.dmp <nazwa_pliku>.fmt
db2trc fmt -c db2trc.dmp <nazwa_pliku>.fmtc
```

Rozdział 23. Pliki śledzenia DRDA

Przed przeanalizowaniem zapisu śledzenia DRDA należy zrozumieć, że architektura DRDA jest otwartym standardem dla definicji struktur danych i komunikacji. Na przykład architektura DRDA obejmuje zestaw reguł dotyczących organizacji danych do przesyłania oraz sposobów komunikacji dla tych informacji. Te reguły są zdefiniowane w następujących podręcznikach:

- DRDA V3 Vol. 1: Distributed Relational Database Architecture
- DRDA V3 Vol. 2: Formatted Data Object Content Architecture
- DRDA V3 Vol. 3: Distributed Data Management Architecture

Wersje PDF tych podręczników są dostępne na stronie www.opengroup.org.

Program narzędziowy **db2drdat** zapisuje dane wymieniane między produktem DRDA Application Requestor (AR) a serwerem DB2 DRDA Application Server (AS) (na przykład między produktem DB2 Connect a hostem lub serwerem bazy danych Series i).

Narzędzie śledzenia

Program narzędziowy db2drdat rejestruje dane wymieniane między serwerem DB2 Connect (w imieniu klienta IBM Data Server Client) a serwerem bazy danych na hoście lub serwerze System i.

Zrozumienie, jak działa ten przepływ danych, może być pożyteczne dla administratora baz danych (lub programisty aplikacji), ponieważ może pomóc w określeniu pochodzenia danego błędu. Na przykład wtedy, gdy zostanie użyta instrukcja bazy danych CONNECT TO w odniesieniu do serwera bazy danych na hoście lub serwerze System i, ale jej wykonanie nie powiedzie się, zostanie zwrócony kod niepowodzenia. Wiedząc dokładnie, jakie informacje były przekazywane do systemu zarządzania serwerem bazy danych na hoście lub serwerze System i, można będzie często określić przyczynę niepowodzenia, nawet jeśli informacje związane z kodem powrotu były ogólne. Wiele niepowodzeń jest spowodowanych przez proste błędy użytkownika.

Dane wyjściowe komendy db2drdat zawierają strumienie danych wymieniane między stacją roboczą DB2 Connect a systemem zarządzania serwerem bazy danych na hoście lub serwerze System i. Dane wysłane do serwera bazy danych na hoście lub serwerze System i są opatrzone etykietą SEND BUFFER, a dane odebrane z hosta lub serwera System i - etykietą RECEIVE BUFFER.

Jeśli bufor odbioru zawiera informacje obszaru komunikacyjnego SQL, wystąpi po nim sformatowana postać tych danych z etykietą SQLCA. Pole SQLCODE obszaru SQLCA jest wartością *nieodwzorowaną* zwróconą przez serwer bazy danych na hoście lub serwerze System i. Bufory wysyłania i odbioru są uporządkowane od najstarszego do najnowszego w pliku. Każdy bufor ma:

- identyfikator procesu,
- etykietę SEND BUFFER, RECEIVE BUFFER lub SQLCA, przy czym pierwsza komenda DDM lub obiekt w buforze ma etykietę DSS TYPE.

Pozostałe dane w buforach wysyłania i odbioru są podzielone na pięć kolumn zawierających:

- liczbę bajtów,

© Copyright IBM Corp. 1993, 2010

- strumień danych DRDA (kolumny 2 i 3) wymienianych między dwoma systemami w kodzie ASCII lub EBCDIC,
- reprezentację kolumn 2 i 3 w kodzie ASCII,
- reprezentację kolumn 2 i 3 w kodzie EBCDIC.

Dane wyjściowe śledzenia

Program narzędziowy db2drdat zapisuje w *pliku śledzenia* następujące informacje:

- -r
  - typ obiektu/odpowiedzi DRDA
  - bufor odbioru
- -s
  - typ żądania DRDA
  - bufor wysyłania
- -c
  - obszar komunikacyjny SQLCA
- Informacje o błędach TCP/IP:
  - kod powrotu funkcji odbioru
  - waga błędu
  - używany protokół
  - używany interfejs API
  - funkcja
  - numer błędu

Uwaga:

1. Wartość zero kodu wyjścia oznacza, że komenda została wykonana pomyślnie, a wartość inna niż zero oznacza, że komenda nie została wykonana pomyślnie.
2. Zwracane są różne pola w zależności od używanego interfejsu API.
3. Zwracane są różne pola w zależności od platformy, na której działa program DB2 Connect, nawet dla tego samego interfejsu API.
4. Jeśli w wyniku komendy db2drdat dane wyjściowe wysyłane są do pliku, który już istnieje, stary plik zostanie usunięty, o ile umożliwiają to uprawnienia do tego pliku.

Analiza pliku danych wyjściowych śledzenia

W pliku śledzenia db2drdat zapisywane są następujące informacje:

- identyfikator procesu (PID, process ID) aplikacji klienta
- nazwa RDB_NAME wpisana do katalogu połączeń bazy danych (DCS)
- identyfikatory CCSID programu DB2 Connect
- identyfikatory CCSID serwera bazy danych na hoście lub serwerze System i
- system zarządzania serwerem bazy danych na hoście lub serwerze System i, z którym komunikuje się system DB2 Connect

Pierwszy bufor zawiera komendy atrybutów serwera wymiany (Exchange Server Attributes, EXCSAT) i dostępu do relacyjnej bazy danych (ACCRDB, Access RDB) wysłane do systemu zarządzania serwerem bazy danych na hoście lub serwerze System i. Są one wysyłane w wyniku działania komendy bazy danych CONNECT TO. Następny bufor zawiera odpowiedzi odebrane przez program DB2 Connect z systemu zarządzania serwerem bazy danych na hoście lub serwerze System i. Zawiera on komendy danych odpowiedzi atrybutów

serwera wymiany (EXCSATRD, Exchange Server Attributes Reply Data) i komunikatu odpowiedzi o dostępie do relacyjnej bazy danych (ACCRDBRM, Access RDB Reply Message).

EXCSAT

Komenda EXCSAT zawiera nazwę stacji roboczej klienta podaną przez obiekt nazwy serwera (SRVNAM, Server Name) o punkcie kodowym X'116D', zgodnie ze specyfikacją DDM. Komenda EXCSAT znajduje się w pierwszym buforze. W ramach komendy EXCSAT wartości X'9481A292' (kodowane w systemie CCSID 500) są tłumaczone na *maskę*, podczas gdy wartość X'116D' zostaje usunięta.

Komenda EXCSAT zawiera także obiekt nazwy zewnętrznej (EXTNAM, External Name), który często jest umieszczany w informacjach diagnostycznych systemu zarządzania baz danych na hoście lub serwerze System i. Składa się on z identyfikatora aplikacji o długości 20 bajtów, po którym następuje identyfikator procesu o długości 8 bajtów (lub identyfikator procesu o długości 4 bajtów i identyfikator wątku o długości 4 bajtów). Jest on reprezentowany przez punkt kodowy X'115E', a w tym przykładzie jego wartość wynosi db2bp i jest dopełniona znakami odstępu z następującą po nich wartością 000C50CC. W produkcie IBM Data Server Client w systemie Linux lub UNIX ta wartość może być związana z komendą ps, która na standardowym wyjściu zwraca informacje o statusie aktywnych procesów.

ACCRDB

Komenda ACCRDB zawiera parametr RDB_NAME w obiekcie RDBNAM o punkcie kodowym X'2110'. Komenda ACCRDB występuje po komendzie EXCSAT w pierwszym buforze. W komendzie ACCRDB wartości X'E2E3D3C5C3F1' są tłumaczone na STLEC1 po usunięciu wartości X'2110'. Odpowiada to polu nazwy docelowej bazy danych w katalogu DCS.

Łańcuch rozliczeniowy ma punkt kodowy X'2104'.

Zestaw kodowy skonfigurowany dla stacji roboczej DB2 Connect można wyświetlić, umieszczając obiekt CCSID: CCSIDSBC (CCSID dla znaków jednobajtowych) o punkcie kodowym X'119C' w komendzie ACCRDB. W podanym przykładzie CCSIDSBC wynosi X'0333', czyli 819.

W komendzie ACCRDB obecne są także obecne dodatkowe obiekty CCSIDDBC (CCSID dla znaków dwubajtowych) i CCSIDMBC (CCSID dla mieszanego zestawu znaków) z punktami kodowymi odpowiednio X'119D' i X'119E'. W podanym przykładzie CCSIDDBC wynosi X'04B0', czyli 1200, a CCSIDMBC wynosi X'0333', czyli 819.

EXCSATRD i ACCRDBRM

Wartości identyfikatorów CCSID są także zwracane z serwera bazy danych na hoście lub serwerze System i w komendzie odpowiedzi na dostęp do RDB (Access RDB Reply Message, ACCRDBRM) w ramach drugiego buforu. Bufor ten zawiera komendę EXCSATRD, po której występuje komenda ACCRDBRM. Przykładowy plik wyjściowy zawiera dwie wartości CCSID dla hosta lub systemu serwera bazy danych System i. Te wartości to 1208 (zarówno dla znaków jednobajtowych, jak i mieszanych) i 1200 (dla znaków dwubajtowych).

Jeśli oprogramowanie DB2 Connect nie rozpozna strony kodowej zwracanej z serwera bazy danych na hoście lub serwerze System i, użytkownik otrzyma kod SQLCODE -332 ze źródłową i docelową stroną kodową. Jeśli serwer bazy danych na hoście lub serwerze System i nie rozpozna zestawu kodowego wysłanego z programu DB2 Connect, zwróci kod o nieobsługiwanej wartości parametru (Parameter Value Not Supported, VALNSPRM) z punktem kodowym DDM X'1252', który dla użytkownika zostanie przetłumaczony na kod SQLCODE -332.

Komenda ACCRDBRM zawiera również parametr PRDID (identyfikator charakterystyczny dla produktu o punkcie kodowym X'112E'). Jego wartość to X'C4E2D5F0F8F0F1F5', co w kodzie EBCDIC odpowiada wartości DSN08015. Zgodnie ze standardami DSN to DB2 Universal Database for z/OS and OS/390. Wskazany jest także numer wersji. ARI to DB2 Server for VSE & VM, SQL to baza danych DB2 lub produkt DB2 Connect, a QSQ to produkt DB2 for i5/OS.

Przykładowy plik danych wyjściowych śledzenia

Poniżej przedstawiono przykładowe dane wyjściowe ilustrujące wybrane strumienie danych DRDA wymieniane między stacjami roboczymi DB2 Connect a serwerem bazy danych na hoście lub serwerze System i. Z punktu widzenia użytkownika komenda bazy danych CONNECT TO została wydana za pośrednictwem procesora procesor wiersza komend (CLP).

Rys. 13 na stronie 145 korzysta z produktu DB2 Connect Enterprise Edition w wersji 9.1 i produktu DB2 Universal Database (UDB) for z/OS w wersji 8 za pośrednictwem połączenia TCP/IP.

```
1 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 0 probe 100
 bytes 16

 Data1  (PD_TYPE_UINT,8) unsigned integer:
 233

2 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 19532 probe 1177
 bytes 250

 SEND BUFFER(AR):

           EXCSAT RQSDSS                    (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  00C3D041000100BD 1041007F115E8482   ...A.....A...^..  .C}........".;db
  0010  F282974040404040 4040404040404040   ...@@@@@@@@@@@@@  2bp
  0020  4040F0F0F0C3F5F0 C3C3F0F0F0000000   @@.............     000C50CC000...
  0030  0000000000000000 0000000000000000   ................  ................
  0040  0000000000000000 000000000060F0F0   ............`..  ............-00
  0050  F0F1A2A495404040 4040404040404040   .....@@@@@@@@@@@  01sun
  0060  4040404040404040 4040404040404040   @@@@@@@@@@@@@@@@
  0070  C4C5C3E5F8404040 F0A2A49540404040   .....@@@....@@@@  DECV8   0sun
  0080  4040404040404040 4000181404140300   @@@@@@@@@.......           .......
  0090  0724070008147400 05240F0008144000   .$....t..$....@.  ..............
  00A0  08000E1147D8C4C2 F261C1C9E7F6F400   ....G....a......  .....QDB2/AIX64.
  00B0  08116D9481A29200 0C115AE2D8D3F0F9   ..m.......Z.....  .._mask...]SQL09
  00C0  F0F0F0                              ...               000

           ACCSEC RQSDSS                    (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  0026D00100020020 106D000611A20003   .&..... .m......  ..}....._...s..
  0010  00162110E2E3D3C5 C3F1404040404040   ..!.......@@@@@@  ....STLEC1
  0020  404040404040                        @@@@@@

3 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110546200 probe 100
 bytes 12

 Data1  (PD_TYPE_UINT,4) unsigned integer:
 105

4 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110549755 probe 1178
 bytes 122

 RECEIVE BUFFER(AR):

           EXCSATRD OBJDSS                  (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  0059D04300010053 1443000F115EE5F8   .Y.C...S.C...^..  ..}.........;V8
  0010  F1C14BE2E3D3C5C3 F100181404140300   ..K.............  1A.STLEC1.......
  0020  0724070007147400 05240F0007144000   .$....t..$....@.  ..............
  0030  0700081147D8C4C2 F20014116DE2E3D3   ....G.......m...  .....QDB2..._STL
  0040  C5C3F14040404040 4040404040000C11   ...@@@@@@@@@@...  EC1          ...
  0050  5AC4E2D5F0F8F0F1 F5                Z........         ]DSN08015

           ACCSECRD OBJDSS                  (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  0010D0030002000A 14AC000611A20003   ..............  ..}.........s..

5 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110656806 probe 100
 bytes 16

 Data1  (PD_TYPE_UINT,8) unsigned integer:
 233
```

Rysunek 13. Przykład pliku wyjściowego śledzenia (połączenie TCP/IP)

```
6 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110659711 probe 1177
 bytes 250

 SEND BUFFER(AR):

           SECCHK RQSDSS                    (ASCII)           (EBCDIC)
          0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  003CD04100010036 106E000611A20003   .<.A...6.n......  ..}......>...s..
  0010  00162110E2E3D3C5 C3F1404040404040   ..!.......@@@@@@  ....STLEC1
  0020  4040404040040000C 11A1D9858799F485   @@@@@@..........        ....Regr4e
  0030  A599000A11A09585 A6A39695           ............      vr....newton

           ACCRDB RQSDSS                    (ASCII)           (EBCDIC)
          0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  00ADD001000200A7 20010006210F2407   ........ ...!.$.  ..}....x........
  0010  00172135C7F9F1C1 F0C4F3C14BD7C1F8   ..!5........K...  ....G91A0D3A.PA8
  0020  F806030221064600 162110E2E3D3C5C3   ....!.F..!......  8..........STLEC
  0030  F140404040404040 4040404040000C11   .@@@@@@@@@@@@...  1            ...
  0040  2EE2D8D3F0F9F0F0 F0000D002FD8E3C4   ............/...  .SQL09000....QTD
  0050  E2D8D3C1E2C30016 00350006119C0333   .........5.....3  SQLASC..........
  0060  0006119D04B00006 119E0333003C2104   ...........3.

7 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 259908001 probe 100
 bytes 12

 Data1  (PD_TYPE_UINT,4) unsigned integer:
 176

8 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 259911584 probe 1178
 bytes 193

 RECEIVE BUFFER(AR):

           SECCHKRM RPYDSS                  (ASCII)           (EBCDIC)
          0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  0015D0420001000F 1219000611490000   ...B.........I..  ..}.............
  0010  000511A400                          .....             ...u.

           ACCRDBRM RPYDSS                  (ASCII)           (EBCDIC)
          0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  009BD00200020095 2201000611490000   ........"....I..  ..}....n........
  0010  000D002FD8E3C4E2 D8D3F3F7F0000C11   .../............  ....QTDSQL370...
  0020  2EC4E2D5F0F8F0F1 F500160035000611   ............5...  .DSN08015.......
  0030  9C04B80006119E04 B80006119D04B000   ................  ................
  0040  0C11A0D5C5E6E3D6 D540400006212524   .........@@..!%$  ...NEWTON  .....
  0050  34001E244E000624 4C00010014244D00   4..$N..$L....$M.  ....+...<.....(.
  0060  06244FFFFF000A11 E8091E768301BE00   .$O........v....  ..!.....Y...c...
  0070  2221030000000005 68B3B8C7F9F1C1F0   "!......h.......  ...........G91A0
  0080  C4F3C1D7C1F8F840 4040400603022106   .......@@@@...!.  D3APA88    .....
  0090  46000A11E8091E76 831389             F......v...       ....Y...c.i

9 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 364420503 probe 100
 bytes 16

 Data1  (PD_TYPE_UINT,8) unsigned integer:
 10
```

Rysunek 14. Przykład pliku wyjściowego śledzenia (połączenie TCP/IP) - kontynuacja

```
10 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 364440751 probe 1177
 bytes 27

 SEND BUFFER(AR):

           RDBCMM RQSDSS                    (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  000AD00100010004 200E               ........ .        ..}.......
11 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 475009631 probe 100
 bytes 12

 Data1  (PD_TYPE_UINT,4) unsigned integer:
 54
12 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 2 nsec 475014579 probe 1178
 bytes 71

 RECEIVE BUFFER(AR):

           ENDUOWRM RPYDSS                  (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  002BD05200010025 220C000611490004   .+.R...%"....I..  ..}.............
  0010  00162110E2E3D3C5 C3F1404040404040   ..!.......@@@@@@  ....STLEC1
  0020  4040404040400005 211501             @@@@@@..!..             .....

           SQLCARD OBJDSS                   (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  000BD00300010005 2408FF             ........$..       ..}........
13 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 721710319 probe 100
 bytes 16

 Data1  (PD_TYPE_UINT,8) unsigned integer:
 126
14 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 721727276 probe 1177
 bytes 143

 SEND BUFFER(AR):

           EXCSQLIMM RQSDSS                 (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  0053D0510001004D 200A00442113E2E3   .S.Q...M ..D!...  ..}....(......ST
  0010  D3C5C3F140404040 4040404040404040   ....@@@@@@@@@@@@  LEC1
  0020  D5E4D3D3C9C44040 4040404040404040   ......@@@@@@@@@@  NULLID
  0030  4040E2D8D3C3F2C6 F0C1404040404040   @@........@@@@@@    SQLC2F0A
  0040  4040404041414141 41484C5600CB0005   @@@@AAAAAAHLV....      ......<.....
  0050  2105F1                              !..               ..1

           SQLSTT OBJDSS                    (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  002BD00300010025 2414000000001B64   .+.....%$......d  ..}.............
  0010  656C657465206672 6F6D206464637375   elete from ddcsu  .%......?_......
  0020  73312E6D79746162 6C65FF             s1.mytable.       ..._`./.%..
15 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 832901261 probe 100
 bytes 12

 Data1  (PD_TYPE_UINT,4) unsigned integer:
 102
```

Rysunek 15. Przykład pliku wyjściowego śledzenia (połączenie TCP/IP) - kontynuacja

```
16 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 832906528 probe 1178
 bytes 119

 RECEIVE BUFFER(AR):

           SQLCARD OBJDSS                    (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  0066D00300010060 240800FFFFFF3434   .f.....`$.....44  ..}....-........
  0010  3237303444534E58 4F544C2000FFFFFE   2704DSNXOTL ....  ......+.!.<.....
  0020  0C00000000000000 00FFFFFFFF000000   ................  ................
  0030  0000000000572020 2057202020202020   .....W   W       ................
  0040  001053544C454331 2020202020202020   ..STLEC1          ....<...........
  0050  2020000F44444353 5553312E4D595441     ..DDCSUS1.MYTA  ............(...
  0060  424C450000FF                        BLE...            .<....
17 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 833156953 probe 100
 bytes 16

 Data1  (PD_TYPE_UINT,8) unsigned integer:
 10
18 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 833159843 probe 1177
 bytes 27

 SEND BUFFER(AR):

           RDBRLLBCK RQSDSS                  (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  000AD00100010004 200F               ........ .        ..}.......
19 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.100)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 943302832 probe 100
 bytes 12

 Data1  (PD_TYPE_UINT,4) unsigned integer:
 54
20 data DB2 UDB DRDA Communication Manager sqljcReceive fnc (3.3.54.3.0.1178)
 pid 807116 tid 1 cpid -1 node 0 sec 5 nsec 943306288 probe 1178
 bytes 71

 RECEIVE BUFFER(AR):

           ENDUOWRM RPYDSS                   (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  002BD05200010025 220C000611490004   .+.R...%"....I..  ..}.............
  0010  00162110E2E3D3C5 C3F1404040404040   ..!.......@@@@@@  ....STLEC1
  0020  4040404040400005 211502             @@@@@@..!..             .....

           SQLCARD OBJDSS                    (ASCII)           (EBCDIC)
         0 1 2 3 4 5 6 7  8 9 A B C D E F   0123456789ABCDEF  0123456789ABCDEF
  0000  000BD00300010005 2408FF             ........$..       ..}........
```

Rysunek 16. Przykład pliku wyjściowego śledzenia (połączenie TCP/IP) - kontynuacja

Informacje w kolejnych buforach dla śledzenia DRDA

Aby uzyskać dodatkowe informacje można analizować kolejne bufory wysyłania i odbioru. W następnym żądaniu występuje instrukcja zatwierdzenia. Komenda commit nakazuje systemowi zarządzania serwerem bazy danych na hoście lub serwerze System i zatwierdzenie bieżącej jednostki pracy. Dane w czwartym buforze są odbierane z systemu zarządzania bazą

danych na hoście lub serwerze System i w wyniku zatwierdzenia lub wycofania. Zawiera on komunikat End Unit of Work Reply Message (ENDUOWRM) oznaczający, że bieżąca jednostka pracy zakończyła się.

W tym przykładzie w pozycji śledzenia 12 występuje pusty obszar SQLCA, co sygnalizuje punkt kodowy DDM X'2408', po którym występuje wartość X'FF'. Obszar komunikacyjny SQL o wartości pustej (X'2408FF') oznacza powodzenie (SQLCODE 0).

Na rysunku Rys. 13 na stronie 145 pokazano przykładowy bufor odbiorczy, w którym na pozycji śledzenia 12 występuje nieprawidłowy obszar SQLCA.

Część 6. Komunikaty

© Copyright IBM Corp. 1993, 2010

Rozdział 24. Najczęściej występujące problemy z produktem DB2 Connect

W tej sekcji opisano najczęstsze objawy problemów z połączeniami, z jakimi można się spotkać podczas korzystania z programu DB2 Connect. W każdym przypadku użytkownik ma do dyspozycji:

- Kombinację numeru komunikatu i kodu powrotu (lub kodu powrotu specyficznego dla używanego protokołu) związanego z tym komunikatem. Każda kombinacja komunikatu i kodu powrotu ma specjalny nagłówek. Nagłówki są uporządkowane według numerów komunikatów, a w następnej kolejności według kodów powrotu.
- Objaw, zwykle w formie wydruku przykładowego komunikatu.
- Sugerowane rozwiązanie wskazujące na prawdopodobną przyczynę błędu. W niektórych sytuacjach dostępnych może być kilka sugerowanych rozwiązań.

SQL0965 lub SQL0969

Objaw Komunikaty SQL0965 i SQL0969 mogą występować z pewną liczbą różnych kodów powrotu zwracanych przez produkty DB2 for i5/OS, DB2 for z/OS oraz DB2 for VM & VSE.

Jeśli zostanie wyświetlony taki komunikat, należy odnaleźć oryginalny kod SQL w dokumentacji dla serwera baz danych wydającego komunikat.

Rozwiązanie

Kodu SQL odebranego z bazy danych na hoście lub serwerze i5/OS nie można konwertować. Należy rozwiązać problem w oparciu o kod błędu, a następnie wprowadzić ponownie komendę, której wykonanie nie powiodło się.

SQL5043N

Objaw Obsługa jednego lub kilku protokołów komunikacyjnych nie została uruchomiona pomyślnie. Jednak podstawowe funkcje menedżera bazy danych zostały uruchomione pomyślnie.

Być może protokół TCP/IP nie został uruchomiony na serwerze DB2 Connect. Wcześniej połączenia z klientem mogły być nawiązywane pomyślnie.

Jeśli diaglevel = 4, dziennik db2diag.log może zawierać pozycje podobne do podanej poniżej.

```
2001-05-30-14.09.55.321092   Instance:svtdbm5   Node:000
PID:10296(db2tcpcm)   Appid:none
common_communication  sqlcctcpconnmgr_child   Probe:46
DIA3205E Adres gniazda "30090" skonfigurowany w pliku
Services TCP/IP
i wymagany przez
obsługę serwera TCP/IP
jest używany przez inny
proces.
```

Rozwiązanie

Ostrzeżenie to oznacza, że program DB2 Connect działający jako serwer dla klientów zdalnych ma trudności z obsługą protokołu komunikacyjnego co najmniej jednego klienta. Mogą to być protokoły TCP/IP lub inne. Komunikat ten zazwyczaj oznacza, że jeden z protokołów komunikacyjnych zdefiniowanych dla programu DB2 Connect nie jest zdefiniowany poprawnie.

© Copyright IBM Corp. 1993, 2010

Często przyczyną może być niezdefiniowana lub niepoprawnie zdefiniowana zmienna profilu DB2COMM. Zazwyczaj problem powstaje w wyniku niezgodności pomiędzy zmienną DB2COMM a nazwami zdefiniowanymi w konfiguracji menedżera bazy danych (na przykład svcename lub nname).

Możliwym scenariuszem jest wykorzystanie połączenia poprzednio zakończonego pomyślnie, a następnie pobranie komunikatu o błędzie SQL5043, gdy żadna konfiguracja nie zmieniła się. Taka sytuacja może mieć miejsce przy użyciu protokołu TCP/IP, gdy zdalny system z jakiegoś powodu niepoprawnie zakończył połączenie. Gdy to nastąpi, połączenie może nadal istnieć po stronie klienta i być może będzie możliwe przywrócenie połączenia po uruchomieniu komend wymienionych poniżej.

Najprawdopodobniej jeden z klientów łączących się z serwerem DB2 Connect nadal ma uchwyt portu TCP/IP. Na każdym komputerze klienckim połączonym z serwerem DB2 Connect należy wykonać komendy:

```
db2 terminate
db2stop
```

SQL30020

Objaw SQL30020N Uruchomienie nie powiodło się z powodu błędu protokołu dystrybucyjnego, który będzie miał wpływ na poprawne uruchomienie kolejnych komend lub instrukcji SQL.

Rozwiązania

Po wystąpieniu tego błędu należy skontaktować się z serwisem. Przed skontaktowaniem się z serwisem należy uruchomić komendę db2support.

SQL30060

Objaw SQL30060N "<*ID autoryzowanego użytkownika*>" nie ma uprawnienia do przeprowadzenia operacji "<*operacja*>".

Rozwiązanie

Przy łączeniu z produktem DB2 for OS/390 and z/OS tabele bazy danych komunikacji (Communications Database - CDB) nie zostały poprawnie zaktualizowane.

SQL30061

Objaw Połączenie z nieprawidłową lokalizacją serwera baz danych na hoście lub serwerze System i - nie można odnaleźć docelowej bazy danych.

Rozwiązanie

Być może w pozycji katalogu DCS podano nieprawidłową nazwę bazy danych serwera. W takiej sytuacji do aplikacji zwracany jest kod SQLCODE -30061.

Sprawdź węzeł DB2, bazę danych i pozycje katalogu DCS. Nazwa docelowej bazy danych w pozycji katalogu DCS musi odpowiadać nazwie bazy danych na danej platformie. W przypadku bazy danych DB2 Universal Database for z/OS and OS/390 używana nazwa powinna być taka sama, jak w polu zestawu danych programu startowego (BSDS) "LOCATION=*nazwa_położenia*". Jest ona także podawana w komunikacie DSNL004I (LOCATION=*położenie*) przy uruchamianiu narzędzia Distributed Data Facility (DDF).

Poprawne komendy dla węzła TCP/IP to:

```
db2 catalog tcpip node <nazwa_węzła> remote <nazwa_lub_adres_hosta>
            server <numer_portu_lub_nazwa_usługi>
db2 catalog dcs database <nazwa_lokalna> as
```

```
                <rzeczywista_nazwa_bazy_danych>
db2 catalog database <nazwa_lokalna> as <alias>
                at node <nazwa_węzła>
              authentication server
```

Następnie, aby połączyć się z bazą danych, wpisz:

```
db2 connect to <alias> user <nazwa_użytkownika> using <hasło>
```

SQL30081N z kodem powrotu 79

Objaw

```
SQL30081N  Wykryto błąd komunikacyjny.
Używany protokół komunikacyjny: "TCP/IP".
Używany komunikacyjny interfejs API: "SOCKETS".
Miejsce,
gdzie wykryto błąd: "".
Funkcja komunikacyjna, która
wykryła błąd:
"connect".
Kody błędów właściwe dla protokołów: "79", "*", "*".
SQLSTATE=08001
```

Rozwiązania

Ten błąd może wystąpić w przypadku, gdy nie powiedzie się połączenie zdalnego klienta z serwerem DB2 Connect. Może wystąpić również przy nawiązywaniu przez serwer DB2 Connect połączenia z hostem lub serwerem bazy danych System i.

1. Zmienna profilu DB2COMM na serwerze DB2 Connect może być nieprawidłowo ustawiona. Należy to sprawdzić. Na przykład przy uruchomionym produkcie DB2 Enterprise Server Edition dla systemu operacyjnego AIX komenda db2set db2comm=tcpip powinna znajdować się w pliku sqllib/db2profile.
2. Specyfikacje nazwy usługi TCP/IP i numeru portu po stronie klienta IBM Data Server Client i serwera DB2 Connect mogą być niezgodne. Sprawdź pozycje w plikach services TCP/IP na obu komputerach.
3. Sprawdź, czy produkt DB2 został uruchomiony na serwerze DB2 Connect. Ustaw wartość 4 dla parametru poziomu diaglevel konfiguracji menedżera bazy danych za pomocą komendy:

   ```
   db2 update dbm cfg using diaglevel 4
   ```

   Po zatrzymaniu i ponownym uruchomieniu programu DB2 sprawdź w pliku db2diag.log, czy uruchomiono komunikację DB2 TCP/IP. Powinien zostać wyświetlony tekst podobny do podanego poniżej:

   ```
   2001-02-03-12.41.04.861119   Instance:svtdbm2   Node:00
   PID:86496(db2sysc)   Appid:none
   common_communication  sqlcctcp_start_listen   Probe:80
   DIA3000I Uruchomienie obsługi protokołu "TCPIP" nie powiodło się.
   ```

SQL30081N z kodem błędu zależnym od protokołu 10032

Objaw

```
SQL30081N  Wykryto błąd komunikacyjny.
Używany protokół komunikacyjny: "TCP/IP".
Używany komunikacyjny interfejs API: "SOCKETS".
Miejsce,
gdzie wykryto błąd : "9.21.85.159".
Funkcja komunikacyjna, która wykryła
błąd: "send".
Kody błędów właściwe dla protokołu: "10032",
"*", "*".
SQLSTATE=08001
```

Rozwiązanie

Ten błąd pojawia się, gdy zachodzi próba odłączenia się od maszyny, na której komunikacja TCP/IP załamała się. Rozwiąż problem z podsystemem TCP/IP.

Na większości komputerów ponowne uruchomienie protokołu TCP/IP likwiduje ten problem. Niekiedy może być konieczne wyłączenie i włączenie komputera.

SQL30082 z kodem powrotu 24 podczas wykonywania instrukcji CONNECT

Objaw SQLCODE -30082 Nieprawidłowa nazwa użytkownika lub hasło.

Rozwiązanie

Należy upewnić się, że w instrukcji CONNECT podano prawidłowe hasło, jeśli jest ono wymagane. Hasło nie jest dostępne do wysłania do bazy danych serwera docelowego. Hasło musi zostać wysłane z klienta IBM Data Server Client do bazy danych na serwerze docelowym. Na niektórych platformach, na przykład AIX, hasło można otrzymać jedynie wtedy, gdy jest ono zawarte w instrukcji CONNECT.

Część 7. Dodatki i uzupełnienia

© Copyright IBM Corp. 1993, 2010

Dodatek A. Przegląd informacji technicznych o programie DB2

Dokumentacja techniczna DB2 jest dostępna za pośrednictwem następujących narzędzi i metod:

- *Centrum informacyjne DB2*
  - tematy (zadanie, opis i informacje dodatkowe)
  - pomoc dotycząca narzędzi DB2
  - programy przykładowe
  - kursy
- podręczniki DB2
  - pliki PDF (możliwe do pobrania)
  - pliki PDF (z dysku DVD z dokumentacją programu DB2 w formacie PDF)
  - podręczniki drukowane
- pomoc dostępna z wiersza komend
  - pomoc dotycząca komend
  - pomoc dotycząca komunikatów

Uwaga: Tematy *Centrum informacyjnego DB2* są aktualizowane częściej niż jakiekolwiek podręczniki w formacie PDF czy w postaci drukowanej. Aby mieć dostęp do najbardziej aktualnych informacji, należy instalować aktualizacje dokumentacji, gdy tylko staną się one dostępne, albo korzystać z *Centrum informacyjnego DB2* w serwisie ibm.com.

Dodatkowe informacje techniczne dla produktu DB2, takie jak noty techniczne, raporty i dokumentacja techniczna IBM Redbooks są dostępne w postaci elektronicznej w serwisie ibm.com. Serwis biblioteki oprogramowania DB2 Information Management jest dostępny pod adresem http://www.ibm.com/software/data/sw-library/.

Opinia na temat dokumentacji

Opinia użytkowników na temat dokumentacji DB2 jest dla nas bardzo ważna. Wszelkie sugestie dotyczące możliwości ulepszenia dokumentacji produktu DB2 można wysyłać pocztą elektroniczną na adres db2docs@ca.ibm.com. Zespół odpowiedzialny za dokumentację DB2 czyta wszystkie nadesłane opinie, ale nie wysyła indywidualnych odpowiedzi. Jeśli jest to możliwe, prosimy o zamieszczenie konkretnych przykładów, które pomogą nam w lepszym zrozumieniu nadesłanej opinii. Jeśli opinia dotyczy konkretnego tematu lub pliku pomocy, należy podać jego tytuł oraz adres URL.

Powyższego adresu email nie należy używać w celu skontaktowania się z działem wsparcia dla Klientów DB2. W przypadku problemów technicznych dotyczących produktu DB2, dla których nie można znaleźć rozwiązania w dokumentacji, w celu uzyskania pomocy należy skontaktować się z lokalnym centrum obsługi IBM.

Wypełnienie ankiety http://www.ibm.com/software/data/info/consumability-survey/ pomoże firmie IBM tworzyć łatwiejsze w obsłudze produkty z rodziny IBM Information Management.

© Copyright IBM Corp. 1993, 2010

Biblioteka techniczna produktu DB2 w formacie PDF lub w postaci drukowanej

W poniższych tabelach opisano bibliotekę DB2 dostępną z centrum publikacji IBM pod adresem www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss. Angielskie i przetłumaczone wersje podręczników produktu DB2 wersja 9.5 w formacie PDF można pobrać z serwisu WWW pod adresem www.ibm.com/support/docview.wss?uid=swg27009727 oraz www.ibm.com/support/docview.wss?uid=swg27009728.

Poniższe tabele zawierają informacje, które z podręczników są dostępne w postaci drukowanej, jednak nie wszystkie z nich mogą być dostępne w danym kraju.

Końcówka numeru zamówienia jest zwiększana po każdej aktualizacji podręcznika. Na podstawie informacji z poniższej tabeli upewnij się, że korzystasz z najnowszych wersji podręczników.

Uwaga: *Centrum informacyjne DB2* jest aktualizowane częściej niż podręczniki PDF i podręczniki drukowane.

Tabela 20. Informacje techniczne dotyczące programu DB2

Nazwa	Numer zamówienia	Dostępne w postaci drukowanej	Ostatnia aktualizacja
Administrative API Reference	SC23-5842-03	Tak	Grudzień, 2010
Administrative Routines and Views	SC23-5843-03	Nie	Grudzień, 2010
Call Level Interface Guide and Reference, Volume 1	SC23-5844-03	Tak	Grudzień, 2010
Call Level Interface Guide and Reference, Volume 2	SC23-5845-03	Tak	Grudzień, 2010
Command Reference	SC23-5846-03	Tak	Grudzień, 2010
Data Movement Utilities Guide and Reference	SC23-5847-03	Tak	Grudzień, 2010
Data Recovery and High Availability Guide and Reference	SC23-5848-03	Tak	Grudzień, 2010
Data Servers, Databases, and Database Objects Guide	SC23-5849-03	Tak	Grudzień, 2010
Database Security Guide	SC23-5850-03	Tak	Grudzień, 2010
Developing ADO.NET and OLE DB Applications	SC23-5851-02	Tak	Kwiecień 2009
Developing Embedded SQL Applications	SC23-5852-02	Tak	Kwiecień 2009
Developing Java Applications	SC23-5853-03	Tak	Grudzień, 2010
Developing Perl and PHP Applications	SC23-5854-02	Nie	Kwiecień 2009

Tabela 20. Informacje techniczne dotyczące programu DB2 (kontynuacja)

Nazwa	Numer zamówienia	Dostępne w postaci drukowanej	Ostatnia aktualizacja
Developing User-defined Routines (SQL and External)	SC23-5855-03	Tak	Grudzień, 2010
Getting Started with Database Application Development	GC23-5856-03	Tak	Grudzień, 2010
DB2 - Wprowadzenie do instalowania i administrowania w środowisku Linux i Windows	GC85-0298-03	Tak	Grudzień, 2010
Internationalization Guide	SC23-5858-02	Tak	Kwiecień 2009
Message Reference, Volume 1	GI11-7855-00	Nie	
Message Reference, Volume 2	GI11-7856-00	Nie	
Podręcznik migracji	GC85-0299-03	Tak	Grudzień, 2010
Net Search Extender Administration and User's Guide	SC23-8509-02	Tak	Kwiecień 2009
Partitioning and Clustering Guide	SC23-5860-03	Tak	Grudzień, 2010
Query Patroller Administration and User's Guide	SC23-8507-01	Tak	Kwiecień 2009
Klienty serwera danych IBM - Krótkie wprowadzenie	GC85-0301-03	Nie	Grudzień, 2010
Serwery DB2 - Krótkie wprowadzenie	GC85-0300-03	Tak	Grudzień, 2010
Spatial Extender and Geodetic Data Management Feature User's Guide and Reference	SC23-8508-02	Tak	Kwiecień 2009
SQL Reference, Volume 1	SC23-5861-03	Tak	Grudzień, 2010
SQL Reference, Volume 2	SC23-5862-03	Tak	Grudzień, 2010
System Monitor Guide and Reference	SC23-5865-03	Tak	Grudzień, 2010
Text Search Guide	SC23-5866-02	Tak	Grudzień, 2010
Troubleshooting Guide	GI11-7857-03	Nie	Grudzień, 2010
Tuning Database Performance	SC23-5867-03	Tak	Grudzień, 2010
Kurs Visual Explain	SC85-0305-00	Nie	
What's New	SC23-5869-03	Tak	Grudzień, 2010

Tabela 20. Informacje techniczne dotyczące programu DB2 (kontynuacja)

Nazwa	Numer zamówienia	Dostępne w postaci drukowanej	Ostatnia aktualizacja
Workload Manager Guide and Reference	SC23-5870-03	Tak	Grudzień, 2010
pureXML Guide	SC23-5871-03	Tak	Grudzień, 2010
XQuery Reference	SC23-5872-02	Nie	Kwiecień 2009

Tabela 21. Informacje techniczne dotyczące produktu DB2 Connect

Nazwa	Numer zamówienia	Dostępne w postaci drukowanej	Ostatnia aktualizacja
DB2 Connect Personal Edition - Krótkie wprowadzenie	GC85-0302-03	Tak	Grudzień, 2010
Serwery DB2 Connect - Krótkie wprowadzenie	GC85-0303-03	Tak	Grudzień, 2010
DB2 Connect - Podręcznik użytkownika	SC85-0304-03	Tak	Grudzień, 2010

Tabela 22. Informacje techniczne dotyczące produktów z rodziny Information Integration

Nazwa	Numer zamówienia	Dostępne w postaci drukowanej	Ostatnia aktualizacja
Information Integration: Administration Guide for Federated Systems	SC19-1020-01	Tak	
Information Integration: ASNCLP Program Reference for Replication and Event Publishing	SC19-1018-02	Tak	
Information Integration: Configuration Guide for Federated Data Sources	SC19-1034-01	Nie	
Information Integration: SQL Replication Guide and Reference	SC19-1030-01	Tak	
Information Integration: Introduction to Replication and Event Publishing	GC19-1028-01	Tak	

Zamawianie drukowanych podręczników z biblioteki DB2

W wielu, choć nie we wszystkich krajach, istnieje możliwość zamówienia drukowanych podręczników DB2 przez Internet. Drukowane podręczniki DB2 można również zamówić u lokalnego przedstawiciela IBM. Należy pamiętać, że nie wszystkie podręczniki umieszczone na dysku DVD *DB2 PDF Documentation* są dostępne w postaci drukowanej. Na przykład żaden z tomów podręcznika *Komunikaty DB2* nie jest dostępny w postaci drukowanej.

Drukowane wersje wielu podręczników DB2 znajdujących się na dysku DVD z dokumentacją DB2 w formacie PDF, można bezpłatnie zamówić w firmie IBM. W zależności od tego skąd

składane jest zamówienie, podręczniki można zamówić przez Internet korzystając z serwisu WWW IBM Publications Center. Jeśli elektroniczne składanie zamówień nie jest dostępne w danym kraju, drukowane podręczniki DB2 można zamówić u lokalnego przedstawiciela IBM. Należy pamiętać, że nie wszystkie podręczniki umieszczone na dysku DVD z dokumentacją DB2 w formacie PDF są dostępne w postaci drukowanej.

Uwaga: Najbardziej aktualna i pełna dokumentacja DB2 znajduje się w Centrum informacyjnym DB2 pod adresem http://publib.boulder.ibm.com/infocenter/db2luw/v9r5.

Aby zamówić drukowane podręczniki DB2:

- Aby sprawdzić czy w danym kraju jest możliwość zamówienia podręczników DB2 poprzez Internet, połącz się z serwisem WWW IBM Publications Center pod adresem http://www.ibm.com/shop/publications/order. Wybierz kraj lub język, aby uzyskać dostęp do informacji na temat zamawiania książek, a następnie postępuj zgodnie z wyświetlanymi instrukcjami.
- Aby zamówić drukowane podręczniki DB2 u lokalnego przedstawiciela IBM:
  1. Odszukaj informacje kontaktowe lokalnego przedstawiciela IBM za pomocą jednego z następujących serwisów WWW:
     - IBM Directory of World Wide Contacts pod adresem www.ibm.com/planetwide
     - IBM Publications pod adresem http://www.ibm.com/shop/publications/order. Wybierz kraj lub język, aby przejść do strony głównej publikacji odpowiedniej dla wybranego kraju lub języka. Na tej stronie kliknij odsyłacz "About this site".
  2. Podczas rozmowy telefonicznej, zaznacz że chcesz zamówić książki DB2 w postaci drukowanej.
  3. Podaj przedstawicielowi IBM tytuły i numery zamówienia podręczników, które chcesz zamówić. Informacje o tytułach i numerach zamówień można znaleźć w sekcji “Biblioteka techniczna produktu DB2 w formacie PDF lub w postaci drukowanej” na stronie 160.

Wyświetlanie pomocy dotyczącej stanu SQL przy użyciu procesora wiersza komend

Program DB2 zwraca wartość SQLSTATE w sytuacji, która mogła zaistnieć na skutek wykonania instrukcji SQL. Pomoc dotycząca wartości SQLSTATE zawiera wyjaśnienia znaczenia stanów SQL i opisy kodów klas stanów SQL.

Aby wyświetlić pomoc dotyczącą stanów SQL, otwórz procesor wiersza komend i wpisz:

```
? stan sql lub ? kod klasy
```

gdzie *stan sql* reprezentuje poprawny pięciocyfrowy stan SQL, a *kod klasy* reprezentuje pierwsze dwie cyfry stanu SQL.
Na przykład: `? 08003` wyświetla pomoc dotyczącą stanu SQL 08003, a `? 08` wyświetli pomoc dotyczącą kodu klasy 08.

Uzyskiwanie dostępu do różnych wersji Centrum informacyjnego DB2

Tematy dotyczące produktu DB2, wersja 9.8, znajdują się w *Centrum informacyjnym DB2* pod adresem http://publib.boulder.ibm.com/infocenter/db2luw/v9r8/.

Tematy dotyczące produktu DB2, wersja 9.7, znajdują się w *Centrum informacyjnym DB2* pod adresem http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/.

Tematy dotyczące produktu DB2, wersja 9.5, znajdują się w *Centrum informacyjnym DB2* pod adresem http://publib.boulder.ibm.com/infocenter/db2luw/v9r5.

Tematy dotyczące produktu DB2, wersja 9.1, znajdują się w *Centrum informacyjnym DB2* pod adresem http://publib.boulder.ibm.com/infocenter/db2luw/v9/.

Tematy dotyczące produktu DB2, wersja 8, znajdują się w *Centrum informacyjnym DB2* pod adresem http://publib.boulder.ibm.com/infocenter/db2luw/v8/.

Wyświetlanie tematów w Centrum informacyjnym DB2 w preferowanym języku użytkownika

Jeśli jest to możliwe, tematy w Centrum informacyjnym DB2 są wyświetlane w języku określonym w preferencjach przeglądarki. Jeśli dany temat nie został przetłumaczony na preferowany język użytkownika, temat ten jest wyświetlany w Centrum informacyjnym DB2 w języku angielskim.

- Aby wyświetlić tematy w preferowanym języku w przeglądarce Internet Explorer:
  1. W programie Internet Explorer kliknij kolejno opcje: **Narzędzia** —> **Opcje internetowe** —> **Języki...**. Zostanie otwarte okno Preferencje językowe.
  2. Sprawdź, czy na liście języków jako pierwszy wyświetlany jest preferowany język.
     - Aby dodać do listy nowy język, kliknij przycisk **Dodaj...**.

       Uwaga: Dodanie języka nie powoduje dodania do systemu czcionek niezbędnych do wyświetlania tematów w preferowanym języku.
     - Aby przenieść język na początek listy, wybierz język i klikaj przycisk **Przenieś w górę** do momentu, gdy język znajdzie się na początku listy.
  3. Wyczyść pamięć podręczną przeglądarki a następnie odśwież stronę, aby wyświetlić Centrum informacyjne DB2 w wybranym języku.
- Aby wyświetlić tematy w wybranym języku w przeglądarce Firefox lub Mozilla:
  1. Naciśnij przycisk w sekcji **Języki** okna dialogowego **Narzędzia** —> **Opcje** —> **Zaawansowane**. W oknie Preferences (Preferencje) zostanie wyświetlony panel Languages (Języki).
  2. Sprawdź, czy na liście języków jako pierwszy wyświetlany jest preferowany język.
     - Aby dodać do listy nowy język, kliknij przycisk **Add... (Dodaj...)**, co pozwoli wybrać język w oknie Add Languages (Dodaj języki).
     - Aby przenieść język na początek listy, wybierz język i klikaj przycisk **Przenieś w górę** do momentu, gdy język znajdzie się na początku listy.
  3. Wyczyść pamięć podręczną przeglądarki a następnie odśwież stronę, aby wyświetlić Centrum informacyjne DB2 w wybranym języku.

W niektórych kombinacjach przeglądarki i systemu operacyjnego może być konieczna zmiana ustawień regionalnych systemu operacyjnego na wybrane ustawienia narodowe i język.

Aktualizowanie Centrum informacyjnego DB2 zainstalowanego na komputerze lokalnym lub serwerze intranetowym

Jeśli Centrum informacyjne DB2 zostało zainstalowane lokalnie, można pobierać i instalować aktualizacje dokumentacji udostępniane przez firmę IBM.

Aby zaktualizować lokalnie zainstalowane *Centrum informacyjne DB2*:

1. Zatrzymaj *Centrum informacyjne DB2* na komputerze i zrestartuj je w trybie autonomicznym. Uruchomienie Centrum informacyjnego w trybie autonomicznym umożliwia jego aktualizację, ale nie pozwala na dostęp do niego innym użytkownikom w sieci. *Centrum informacyjne DB2* użytkowników innych niż administrator i root zawsze uruchamiane jest w trybie autonomicznym. .
2. Aby sprawdzić dostępne aktualizacje, należy użyć funkcji aktualizacji. Jeśli są aktualizacje, które chcesz zainstalować, opcja aktualizacji umożliwia ich pobranie i zainstalowanie.

   Uwaga: Jeśli dane środowisko wymaga zainstalowania aktualizacji *Centrum informacyjnego DB2* na komputerze, który nie jest podłączony do sieci, należy utworzyć lokalną kopię ośrodka aktualizacyjnego za pomocą komputera podłączonego do sieci i z zainstalowanym *Centrum informacyjnego DB2*. Jeśli w danej sieci aktualizację będzie instalować wielu użytkowników, można skrócić czas wymagany do zainstalowania aktualizacji na komputerze danej osoby, tworząc lokalną kopię ośrodka aktualizującego i konfigurując dla niego serwer proxy.
   Jeśli pakiety aktualizacyjne są dostępne, należy je pobrać za pomocą opcji aktualizacji. Opcja aktualizacji jest dostępna tylko w trybie autonomicznym.
3. Zatrzymaj autonomiczne Centrum informacyjne i zrestartuj *Centrum informacyjne DB2* na komputerze.

Uwaga: W systemie Windows Vista wymienione niżej komendy muszą być uruchamiane przez administratora. Aby uruchomić wiersz komend lub program graficzny z pełnymi uprawnieniami administratora, kliknij skrót prawym przyciskiem myszy i wybierz opcję **Uruchom jako**.

Aby zaktualizować *Centrum informacyjne DB2* zainstalowane na komputerze lub serwerze intranetowym:

1. Zatrzymaj *Centrum informacyjne DB2*.
   - W systemie Windows kliknij kolejno opcje: **Start → Panel sterowania → Narzędzia administracyjne → Usługi**. Następnie kliknij prawym przyciskiem myszy usługę **Centrum informacyjne** DB2 i wybierz opcję **Zatrzymaj**.
   - W systemie Linux wprowadź następującą komendę:
     ```
     /etc/init.d/db2icdv95 stop
     ```
2. Uruchom Centrum informacyjne w trybie autonomicznym.
   - W systemie Windows:
     a. Otwórz okno komend.
     b. Przejdź do katalogu, w którym zainstalowane jest Centrum informacyjne. Domyślnie *Centrum informacyjne DB2* jest instalowane w katalogu *Program_files*\IBM\DB2 Information Center\Version 9.5, gdzie *Program_files* reprezentuje położenie katalogu Program Files.
     c. Przejdź z katalogu instalacyjnego do podkatalogu doc\bin.
     d. Uruchom plik help_start.bat:
        ```
        help_start.bat
        ```
   - W systemie Linux:
     a. Przejdź do katalogu, w którym zainstalowane jest Centrum informacyjne. Domyślnie *Centrum informacyjne DB2* jest instalowane w katalogu /opt/ibm/db2ic/V9.5.
     b. Przejdź z katalogu instalacyjnego do podkatalogu doc/bin.
     c. Uruchom skrypt help_start:
        ```
        help_start
        ```

Zostanie uruchomiona domyślna przeglądarka WWW systemu, a w niej autonomiczne Centrum informacyjne.

3. Kliknij przycisk **Aktualizuj** ( ). W prawym panelu Centrum informacyjnego kliknij opcję **Znajdź aktualizacje**. Zostanie wyświetlona lista aktualizacji dla istniejącej dokumentacji.
4. Aby zainicjować proces instalowania, zaznacz elementy, które chcesz zainstalować, a następnie kliknij przycisk **Zainstaluj aktualizacje**.
5. Po zakończeniu procesu instalowania, kliknij przycisk **Zakończ**.
6. Zatrzymaj autonomiczne Centrum informacyjne:
   - W systemach Windows, przejdź do podkatalogu doc\bin w katalogu instalacyjnym i uruchom plik help_end.bat:
     ```
     help_end.bat
     ```
     Uwaga: Plik wsadowy help_end zawiera komendy wymagane do bezpiecznego zakończenia procesów, które zostały uruchomione za pomocą pliku wsadowego help_start. Do zakończenia działania pliku help_start.bat nie należy używać kombinacji klawiszy Ctrl-C lub jakiejkolwiek innej metody.
   - W systemie Linux, przejdź do podkatalogu doc/bin w katalogu instalacyjnym i uruchom skrypt help_end:
     ```
     help_end
     ```
     Uwaga: Skrypt help_end zawiera komendy wymagane do bezpiecznego zakończenia procesów, które zostały uruchomione za pomocą skryptu help_start. Do zakończenia działania skryptu help_start nie należy używać żadnej innej metody.
7. Zrestartuj *Centrum informacyjne DB2*.
   - W systemie Windows kliknij kolejno opcje: **Start → Panel sterowania → Narzędzia administracyjne → Usługi**. Następnie kliknij prawym przyciskiem myszy usługę **Centrum informacyjne** DB2 i wybierz opcję **Uruchom**.
   - W systemie Linux wprowadź następującą komendę:
     ```
     /etc/init.d/db2icdv95 start
     ```

Zostanie wyświetlone zaktualizowane *Centrum informacyjne DB2* z nowymi, zaktualizowanymi tematami.

Kursy DB2

Kursy DB2 pomagają w poznawaniu różnych aspektów dotyczących produktów DB2. Lekcje zawierają szczegółowe instrukcje krok po kroku.

Zanim rozpoczniesz

Kurs w wersji XHTML można wyświetlić w Centrum informacyjnym pod adresem: http://publib.boulder.ibm.com/infocenter/db2help/.

W niektórych lekcjach wykorzystano przykładowe dane lub kod programu. Opisy wymagań wstępnych dla wykonania konkretnych zadań znajdują się w treści kursu.

Kursy DB2

Aby wyświetlić kurs, kliknij jego tytuł.

“pureXML” w *pureXML Guide*
Konfigurowanie bazy danych DB2 do przechowywania danych XML i przeprowadzania podstawowych operacji na składnicy rodzimych danych XML.

“Visual Explain” w *Kurs Visual Explain*
Poprawianie wydajności drogą analizy, optymalizacji i dostrajania instrukcji SQL za pomocą programu Visual Explain.

Informacje dotyczące rozwiązywania problemów z programem DB2

Podczas korzystania z produktów bazodanowych DB2 użytkownik ma do dyspozycji wiele różnych informacji dotyczących diagnozowania i rozwiązywania problemów.

Dokumentacja DB2
Informacje na temat rozwiązywania problemów można znaleźć w podręczniku DB2 Troubleshooting Guide lub w sekcji "Podstawy baz danych" w Centrum informacyjnym DB2. Znajdują się tam informacje na temat wyodrębniania i identyfikowania problemów przy użyciu diagnostycznych narzędzi DB2, opisy rozwiązań najczęstszych problemów, oraz inne porady dotyczące rozwiązywania problemów, które mogą wystąpić podczas korzystania z produktów bazodanowych DB2.

Serwis WWW wsparcia technicznego dla programu DB2
W razie wystąpienia problemów i konieczności uzyskania pomocy w znalezieniu prawdopodobnych przyczyn i możliwych rozwiązań, należy odwiedzić serwis WWW wsparcia technicznego dla programu DB2. Serwis ten zawiera odsyłacze do najnowszych publikacji dotyczących programu DB2, not technicznych, raportów APAR (Authorized Program Analysis Reports), pakietów poprawek i innych zasobów. Użytkownik może przeszukiwać tę bazę wiedzy, aby znaleźć możliwe rozwiązania określonych problemów.

Serwis WWW wsparcia technicznego dla programu DB2 jest dostępny pod adresem: http://www.ibm.com/software/data/db2/support/db2_9/

Warunki

Zezwolenie na korzystanie z tych publikacji jest przyznawane na poniższych warunkach.

Użytek osobisty: Użytkownik ma prawo kopiować te publikacje do własnego, niekomercyjnego użytku pod warunkiem zachowania wszelkich uwag dotyczących praw własności. Użytkownik nie ma prawa dystrybuować ani wyświetlać tych publikacji czy ich części, ani też wykonywać na ich podstawie prac pochodnych bez wyraźnej zgody IBM.

Użytek służbowy: Użytkownik ma prawo kopiować te publikacje, dystrybuować je i wyświetlać wyłącznie w ramach przedsiębiorstwa Użytkownika pod warunkiem zachowania wszelkich uwag dotyczących praw własności. Użytkownik nie ma prawa wykonywać na podstawie tych publikacji ani ich części prac pochodnych, kopiować ich, dystrybuować ani wyświetlać poza przedsiębiorstwem Użytkownika bez wyraźnej zgody IBM.

Z wyjątkiem zezwoleń wyraźnie udzielonych w niniejszym dokumencie, nie udziela się jakichkolwiek innych zezwoleń, licencji ani praw, wyraźnych czy domniemanych, odnoszących się do tych publikacji czy jakichkolwiek informacji, danych, oprogramowania lub innej własności intelektualnej, o których mowa w niniejszym dokumencie.

IBM zastrzega sobie prawo do anulowania zezwolenia przyznanego w niniejszym dokumencie w każdej sytuacji, gdy, według uznania IBM, korzystanie z tych publikacji jest szkodliwe dla IBM lub jeśli IBM uzna, że warunki niniejszego dokumentu nie są przestrzegane.

Użytkownik ma prawo pobierać, eksportować lub reeksportować niniejsze informacje pod warunkiem zachowania bezwzględnej i pełnej zgodności z obowiązującym prawem i przepisami, w tym ze wszelkimi prawami i przepisami eksportowymi Stanów Zjednoczonych.

IBM NIE UDZIELA JAKICHKOLWIEK GWARANCJI, W TYM TAKŻE RĘKOJMI, DOTYCZĄCYCH TREŚCI TYCH PUBLIKACJI. PUBLIKACJE TE SĄ DOSTARCZANE W STANIE, W JAKIM SIĘ ZNAJDUJĄ ("AS-IS") BEZ UDZIELANIA JAKICHKOLWIEK GWARANCJI, W TYM TAKŻE RĘKOJMI, WYRAŹNYCH CZY DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ CZY PRZYDATNOŚCI DO OKREŚLONEGO CELU.

Dodatek B. 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 w innych 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
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 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 JAKICHKOLWIEK GWARANCJI (W TYM TAKŻE RĘKOJMI), WYRAŹNYCH 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.

© Copyright IBM Corp. 1993, 2010

Niniejszy dokument może zawierać odsyłacze lub odniesienia do serwisów WWW i zasobów stron trzecich. IBM nie dokonuje jakichkolwiek ustaleń, nie udziela gwarancji (w tym także rękojmi) i nie podejmuje jakichkolwiek zobowiązań dotyczących serwisów WWW nienależących do IBM oraz zasobów stron trzecich, które mogą być przywołane w niniejszym dokumencie, z niego dostępne bądź do których odsyłacz tenże dokument może zawierać. Odsyłacz do serwisu WWW strony trzeciej nie oznacza, że IBM wyraża swoją zgodę na zawartość lub używanie takiego serwisu WWW bądź że udziela poparcia jego właścicielowi. Ponadto IBM nie jest stroną jakichkolwiek transakcji, które użytkownik może zawrzeć ze stronami trzecimi, ani też nie ponosi za takie transakcje odpowiedzialności, nawet jeśli dowiedział się o takich stronach trzecich (lub używał odsyłacza do ich serwisu) za pośrednictwem serwisu WWW IBM. W związku z tym użytkownik potwierdza i zgadza się ze stwierdzeniem, że IBM nie ponosi odpowiedzialności za dostępność takich zewnętrznych serwisów WWW czy zasobów ani też za treści, usługi, produkty bądź inne materiały znajdujące się w takich serwisach i zasobach bądź z nich dostępne. Wszelkie oprogramowanie dostarczane przez strony trzecie podlega warunkom licencji dołączonych to takiego oprogramowania.

IBM ma prawo do korzystania i rozpowszechniania informacji przysłanych przez użytkownika w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

Licencjobiorcy 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 Canada
Limited
U59/3600
3600 Steeles Avenue East
Markham, Ontario L3R 9Z7
CANADA

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.

Jakiekolwiek wzmianki na temat kierunków rozwoju firmy IBM mogą ulec zmianie lub anulowaniu bez uprzedzenia i dotyczą jedynie ogólnych celów i założeń.

Publikacja ta może zawierać 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 może zawierać 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 uiszczania opłat na rzecz firmy IBM, 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.

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 IBM Corp. Sample Programs. © Copyright IBM Corp. *_rok_lub_lata_*. Wszelkie prawa zastrzeżone.

Znaki towarowe

IBM, logo IBM i ibm.com są znakami towarowymi lub zastrzeżonymi znakami towarowymi International Business Machines Corp. w Stanach Zjednoczonych i/lub innych krajach. Inne nazwy produktów i usług mogą być znakami towarowymi IBM lub innych firm. Aktualna lista znaków towarowych IBM jest dostępna w sieci WWW, w sekcji Copyright and trademark information na stronie www.ibm.com/legal/copytrade.shtml.

Poniższe nazwy są znakami towarowymi lub zastrzeżonymi znakami towarowymi innych firm:

- Linux jest zastrzeżonym znakiem towarowym Linusa Torvaldsa w Stanach Zjednoczonych i/lub innych krajach.
- Java i wszystkie znaki towarowe oraz logo związane z Java są znakami towarowymi Sun Microsystems, Inc. w Stanach Zjednoczonych i/lub innych krajach.
- UNIX jest zastrzeżonym znakiem towarowym The Open Group w Stanach Zjednoczonych i innych krajach.
- Intel, logo Intel, Intel Inside, logo Intel Inside, Intel Centrino, logo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium i Pentium są znakami towarowymi lub zastrzeżonymi znakami towarowymi Intel Corporation lub przedsiębiorstw podporządkowanych w Stanach Zjednoczonych i/lub innych krajach.
- Microsoft, Windows, Windows NT i logo Windows są znakami towarowymi Microsoft Corporation w Stanach Zjednoczonych i/lub innych krajach.

Inne nazwy firm, produktów i usług mogą być znakami towarowymi lub znakami usług innych firm.

Indeks

A

ACCRDB, komenda 142
ACCRDBRM, komenda 142
ACCSEC, komenda 142
agentpri, parametr konfiguracyjny menedżera baz danych 117
aktualizacje
  Centrum informacyjne DB2 164
  katalog bazy danych 31
aktualizacje na wielu serwerach
  Centrum sterowania 56
  menedżer punktów synchronizacji 57
  rozproszona jednostka pracy (DUOW) 55
  testowanie 56
  włączanie 55
alerty poprawności 76
alias bazy danych klienta 72
ampersand (podwójny &)
  plik odwzorowania SQLCODE 63
aplikacje
  procedury składowane 101
  projektowanie 101
  wiązanie 51
  WWW
    korzystanie z programu DB2 Connect 20
  wydajność 101
  złożony SQL 101
aplikacje klienckie
  identyfikator 72
  odtwarzanie komunikacji 89
aplikacje WWW
  DB2 Connect 20
  procedury składowane 22
arkusze robocze
  dostosowywanie katalogu 37
automatyczne przekierowanie klienta
  awarie połączenia 91
  konfiguracja dla produktu DB2 Connect 89
  opis 89
awarie połączenia
  automatyczne przekierowanie klienta 91

B

bazy danych
  aliasy
    arkusz dostosowania katalogu 37
    systemowy katalog baz danych 31
  bazy danych hosta 6
  grupowanie żądań 101
  narzędzia do pomiaru wydajności 95
  nazwy
    arkusz dostosowania katalogu 37
    DCS, katalog 33
    RDBNAM, obiekt 142
    systemowy katalog baz danych 31
  strojenie 118
bazy danych hosta
  dostęp przy użyciu DB2 Connect Personal Edition 15
  połączenia
    równoważenie obciążenia 87
bazy danych hosta *(kontynuacja)*
  połączenia *(kontynuacja)*
    wysoka dostępność 87
bezpieczeństwo
  DB2 Connect
    obsługa 49
  GRANT, instrukcja 49
  Kerberos 47
  kody rozszerzone
    OS/390 i z/OS 48
  podpowiedzi 48
  REVOKE, instrukcja 49
  TCP/IP 49
  typy 37
  wartości w katalogu węzłów 32
  wprowadzenie 41
  wskazówki 48
BINDADD, uprawnienie
  uprawnienie do wiązania 51
bloki zapytań
  zwiększanie szybkości przesyłania danych przez oprogramowanie DB2 Connect 121
błędy
  rozwiązywanie problemów 131
bufor odbioru 141
bufor wysyłania
  śledzenie danych 141

C

CCSID (identyfikator kodowanego zestawu znaków)
  obsługa języków dwukierunkowych
    opis 33
Centrum informacyjne DB2
  aktualizowanie 164
  języki 164
  wersje 163
  wyświetlanie w różnych językach 164
Centrum sterowania
  aktualizacje na wielu serwerach 56
CGI (Common Gateway Interface), programowanie
  ograniczenia 20
  zalety 20
CHAR, typ danych
  opis 124
Character Data Representation Architecture (CDRA) 9
CLI (Call Level Interface)
  aplikacje
    CURRENTPACKAGESET 48
  przegląd 127
  zaufane połączenia 41
COMMIT, instrukcja
  statycznie powiązana 101
commit, komenda
  śledzenie buforów wyjściowych 142
Common Gateway Interface (CGI), programowanie
  ograniczenia 20
  zalety 20
core, pliki
  diagnozowanie problemów 134
CREATE IN COLLECTION NULLID, uprawnienie 51

© Copyright IBM Corp. 1993, 2010

CURRENTPACKAGESET CLI/ODBC, parametr 48
czas odpowiedzi 95

D

D (rozłączenie), parametr 33
dane
  łączenie w bloki 101
  przepływy
    DB2 Connect 9, 95
  przesyłanie
    między hostami a stacjami roboczymi 59
    prędkości 95, 124
    wydajność 124
  źródła
    żądanie rozproszone 12
daty
  obsługa stref czasowych 33
DB2 Connect
  bezpieczeństwo 41
  Enterprise Edition
    API 21
    aplikacje WWW 20
    JDBC 21
    menedżery transakcji zgodne z XA 57
    monitory przetwarzania transakcyjnego 26
    serwery połączeń 18
    serwery WWW 22
    SQLJ 21
  koncentratory połączeń 112
  obsługa hosta 15
  obsługa produktu System i 15
  obsługa systemu Sysplex 113
  produkty 3
  przegląd 3
  przenoszenie danych 59
  scenariusze dotyczące serwerów połączeń 15
  udoskonalenia 3
DB2 Universal Database for OS/390 and z/OS
  bezpieczeństwo 48
  BSDS, parametry 32
  DOMAIN 32
  DYNAMICRULES (BIND), opcja 48
  RESPORT 32
  TCPPORT 32
  wartości w katalogu węzłów 32
  zestaw danych programu startowego 32
db2drdat, komenda
  plik wyjściowy 141
db2trc, komenda
  formatowanie danych wyjściowych śledzenia 138
  przegląd 137
  zrzut danych wyjściowych śledzenia 138
DCS, katalog dostępu do bazy danych
  katalog dostępu do bazy danych (DCS) 33
dcs1ari.map, plik 63
dcs1dsn.map, plik 63
dcs1qsq.map, plik 63
ddcs400.lst, plik 51
ddcsmvs.lst, plik 51
ddcstrc, program narzędziowy 142
ddcsvm.lst, plik 51
ddcsvse.lst, plik 51
DDM (Distributed Data Management)
  architektura DRDA 9
  wynik komendy db2drdat 141
DESCRIBE, instrukcja
  wydajność a instrukcja PREPARE 101
  złożone instrukcje SQL 101
diagnozowanie problemów
  dostępne informacje 167
  kursy 167
  narzędzia diagnostyczne
    przegląd 134
  po nawiązaniu połączenia 132
  połączenie 131
  problemy po nawiązaniu połączenia 133
DIRCACHE, parametr 117
Distributed Data Management (DDM) 133
  Distributed Relational Database Architecture (DRDA) 9
  wynik komendy db2drdat 141
Distributed Relational Database Architecture (DRDA)
  dostęp do danych 9
  przegląd 9
docelowe bazy danych
  nazwa 33, 37
dodatkowe bloki zapytania
  EXTRA BLOCKS SRV, parametr 122
  przegląd 122
dokumentacja
  drukowana 160
    zamawianie 162
  PDF 160
  przegląd 159
  warunki używania 167
dostosowywanie
  katalogi, arkusze 37
dynamiczny SQL
  CURRENTPACKAGESET CLI/ODBC, parametr
    konfiguracyjny 48
  wydajność
    sposoby poprawiania 101
  wyniki przetwarzania 6
dziennik wartościowania strategii
  monitor poprawności DB2 for z/OS 76

E

ENDUOWRM, komunikat 142
exchange server attributes, komenda 142
EXCSAT, komenda 142
EXCSATRD, komenda 142
EXECUTE IMMEDIATE, instrukcja
  projektowanie aplikacji 101
EXTNAM, obiekt 142

F

FOR FETCH ONLY, klauzula
  SELECT, instrukcja 101
FORCE, komenda
  identyfikator agenta 72
Formatted Data Object Content Architecture (FDOCA) 9

G

GET SNAPSHOT, komenda 70
GRANT, instrukcja
  bezpieczeństwo 49

H

hasła
  zmiana
    OS/390 i z/OS 48

I

IBM WebSphere 21
ID autoryzowanego użytkownika 72
identyfikator aplikacji hosta 72
identyfikator produktu klienta 72
identyfikator strony kodowej 72
informacje diagnostyczne
  przegląd 134
instrukcje
  COMMIT 101
  DESCRIBE 101
  EXECUTE IMMEDIATE 101
  FOR FETCH ONLY 101
  GRANT 49
  PREPARE 101
  REVOKE 49
  ROLLBACK
    projektowanie aplikacji 101
  SELECT 101
instrukcje SQL
  DB2 Connect 6
  wyświetlanie pomocy 163
INTEGER, typ danych 123
INTERRUPT_ENABLED (rozłączenie), parametr 33
iSeries
  DRDA 9

J

Java
  serwery aplikacji
    API 21
    DB2 Connect 21
    JDBC 21
    SQLJ 21
jednostki pracy (UOW)
  definicja 9
  rozproszone 55
  zdalne 11

K

katalog bazy danych
  aktualizowanie 31
  Database Connection Services (DCS) 31
  węzeł 31
  wiele pozycji 38
katalog dostępu do bazy danych (DCS)
  aktualizowanie wpisów 31
  wartości 33
katalogi
  arkusze dostosowania 37
  systemowa baza danych
    aktualizowanie 31
    wartości 31
kolejny numer wychodzący
  DB2 LIST DCS APPLICATIONS, komenda 72
komendy
  ACCRDB 142
komendy *(kontynuacja)*
  ACCRDBRM 142
  ACCSEC 142
  commit 142
  db2drdat 141
  db2trc 137, 138
  EXCSAT 142
  EXCSATRD 142
  GET SNAPSHOT 70
  SECCHK 142
komunikaty o błędach
  DB2 Connect 153
koncentrator połączeń 105
koncentratory połączeń
  agenty logiczne 107
  agenty pracujące 107
  DB2 Connect 112
  implementacja 107
  łączenie w pule 107
  MAX_COORDAGENTS, parametr konfiguracyjny 107
  narzut 107
  NUM_INITAGENTS, parametr konfiguracyjny 107
  NUM_POOLAGENTS, parametr konfiguracyjny 107
  obsługa transakcji XA 107
  ograniczenia 107
  parametry konfiguracyjne 107
  porównanie z zestawianiem połączeń 111
  program rozsyłający 107
  przegląd 105
  przykłady 107
konfigurowanie
  połączenia z hostem 15
  uwagi, zmiana hasła 48
konwersja danych
  host 123
konwersje
  dane hosta 123
Kreator konfiguracji aktualizacji na wielu serwerach 56
kreatorzy
  aktualizacja na wielu serwerach 56
kursy
  diagnozowanie problemów 167
  rozwiązywanie problemów 167
  Visual Explain 166

L

LIST DCS APPLICATIONS, komenda 72
lista adresów w pamięci podręcznej 114
lista powiązań 51
LOCALDATE, parametr 33

Ł

łańcuchy parametrów
  podwójne przecinki 33
  przecinki 33
łączenie danych w bloki 101

M

max_coordagents, parametr konfiguracyjny menedżera baz danych 105, 107
maxagents, parametr konfiguracyjny menedżera baz danych 117
MAXDARI, parametr konfiguracyjny 117

menedżer punktów synchronizacji (SPM)
parametry domyślne 57
scenariusze 57
menedżery transakcji XA
koncentratory połączeń 107
opis 26
metoda dostępu do sieci telekomunikacyjnej (VTAM) 49
Microsoft Windows
aplikacje 15
monitor poprawności
DB2 for z/OS 76
monitor poprawności DB2 for z/OS
przegląd 76
przeglądanie, wprowadzanie, zapisywanie zalecanych działań 78
uruchamianie, zatrzymywanie, odświeżanie 77
wyświetlanie obiektów alertów 82
wyświetlanie podsumowań alertów 81
monitor systemu baz danych
klienty zdalne 69
przegląd 7
monitorowanie
Monitor wydajności Windows 69
połączenia 69
serwer DB2 Connect 69
wydajność 69
monitory przetwarzania transakcyjnego
aktualizacje na wielu serwerach 55
charakterystyki wykorzystania 26
OLTP 26
przykłady 26
transakcje 26
Tuxedo 26

N

narzędzia
wydajność 95
wykorzystanie jednostki centralnej 95
wykorzystanie pamięci 95
narzędzia wykorzystujące pamięć 95
narzędzie eksportujące
przesyłanie danych między hostem a stacją roboczą 59
narzędzie importujące
przesyłanie danych między hostem a stacją roboczą 59
narzędzie śledzenia
pliki śledzenia DB2 137, 138
przegląd 137
śledzenie DRDA 144, 148
narzędzie śledzenia (db2drdat) 141
nazwa aplikacji, element monitora 72
nazwa docelowej bazy danych serwera AS 33
nieupakowane dziesiętne
typy danych 123
NOMAP, parametr
odwzorowanie SQL CODE 33
parametry katalogu DCS 63
wyłączanie odwzorowania kodów SQL 63
NULLID
OS/400 51
num_initagents, parametr konfiguracyjny menedżera baz danych 105, 107
num_poolagents, parametr konfiguracyjny menedżera baz danych 105, 107
NUMDB
parametr 117
numer kolejny klienta 72

O

o tym podręczniku vii
obiekty alertów
wyświetlanie 82
obsługa identyfikatorów CCSID dla języków dwukierunkowych
parametr BIDI 33
ODBC (Open Database connectivity)
aplikacje
CURRENTPACKAGESET 48
interfejs 15
optymalizacja dostępu 99
przegląd 127
odświeżanie
monitor poprawności DB2 for z/OS 77
odtwarzanie komunikacji
aplikacje klienckie 89
odwołania
definiowanie wielu wpisów bazy danych 38
odwzorowanie
kody SQLCODE 63
NOMAP, parametr 63
odwzorowanie kodów SQLCODE
dostosowywanie 63
NOMAP, parametr 63
ograniczenia
koncentrator połączeń 107
OS/390
DRDA 9

P

pakiety
serwery bazy danych hosta 51
serwery bazy danych System i 51
parametr konfiguracyjny obsługi pamięci podręcznej katalogu
Dostrajanie programu DB2 Connect 117
parametr NNAME klienta 72
parametry
katalogi 37
łańcuchy 38
MAXAGENTS 117
PRDID 142
SYSPLEX 33
parametry konfiguracyjne
agentpri 117
DIRCACHE 117
max_coordagents 105, 107
MAXDARI 117
num_initagents 105, 107
num_poolagents 105, 107
numdb 117
rqrioblk 117
TCP_KEEPALIVE 91
plik odwzorowania SQLCODE 63
podsumowania alertów
wyświetlanie 81
połączenia
bezpośrednie z bazą danych hosta lub systemu System i 17
bezpośrednie z hostem 15
DB2 Connect Enterprise Edition 18
koncentratory, patrz koncentratory połączeń 107
łączenie w pule
koncentratory połączeń 107
przegląd 105
zalety 107

połączenia *(kontynuacja)*
ponowne nawiązywanie
bezpośrednie z hostem 15
DB2 Connect Enterprise Edition 18
serwery, DB2 Connect Enterprise Edition 18
pomoc
instrukcje SQL 163
wybieranie preferowanego języka 164
PRDID, parametr 142
predykaty
wydajność logiki 101
PREPARE, instrukcja
projektowanie aplikacji 101
wpływ na wydajność 101
procedury składowane
przegląd 22
procesor
narzędzia do pomiaru wykorzystania 95
procesor wiersza komend (CLP)
instrukcje SQL 7
wydajność 101
program narzędziowy do administracji
DB2 Connect 7
programy narzędziowe
administrowanie, DB2 Connect 7
db2drdat 141
ddcspkgn 51
monitor systemu baz danych 7
ps (status procesu) 134, 142
status procesu 142
śledzenie 141
wiązanie 51
protokoły
wartościowanie strategii 76
protokół uwierzytelniania Kerberos
DB2 Connect 46
OS/390 i z/OS 47
przenoszenie danych
korzystanie z programu DB2 Connect 59
przepustowość
transakcje 95
przykłady
koncentratory połączeń 107
koncentratory XA 107
ps (status procesu), narzędzie 134, 142
ps, komenda
EXTNAM, obiekt 142
przegląd 134

R

RACF, opcja
bezpieczeństwo 49
relacje zaufania
zaufane konteksty i zaufane połączenia 41
Relational Connect
opis produktu 7
requestery aplikacji (AR)
definicja DRDA 9
parametry 37
REVOKE, instrukcja
bezpieczeństwo 49
ROLLBACK, instrukcja
statycznie powiązana 101
rozproszona jednostka pracy
aktualizacje na wielu serwerach 55
obsługiwane serwery 55
rozproszona jednostka pracy *(kontynuacja)*
przegląd 9
zatwierdzanie dwufazowe 55
rozszerzenie wersji 3
rozwiązywanie problemów
DB2 Connect 153
informacje w postaci elektronicznej 167
kursy 167
narzędzia śledzenia 137
DRDA 144, 148
połączenie 131, 132
przegląd 131
wydajność 121
zbieranie informacji 131
znane problemy 133
RQRIOBLK, parametr
strojenie 117
rywalizacja
zasoby systemowe 120

S

scenariusze
zabezpieczenia TCP/IP 49
SECCHK, komenda 142
SELECT, instrukcja
FOR FETCH ONLY, opcja włączona 101
w projekcie aplikacji 101
z możliwością aktualizacji 101
serwery
aplikacja
DB2 Connect EE 23
serwery aplikacji (AS)
definicja DRDA 9
klienty pełnowymiarowe 23
konfigurowanie 23
modele dwuwarstwowe i trójwarstwowe 23
obsługa w programie DB2 Connect 23
przegląd 23
rozmieszczanie 23
serwery WWW
DB2 Connect 22
SET CURRENT PACKAGESET, instrukcja 48
SHOW DETAIL, opcja monitora 72
sieć
narzędzia do pomiaru wydajności 95
strojenie 119
szybkość przesyłania danych 124
skalowanie okna
rozszerzenia RFC-1323 123
SOCKS
węzeł
obowiązkowe zmienne środowiskowe 32
sprzęt
wydajność sieci 124
SQL (Structured Query Language)
dynamiczny 101
statyczny 101
SQL/DS
DRDA 9
SQL_ATTR_
TRUSTED_CONTEXT_PASSWORD
zastosowanie 43
TRUSTED_CONTEXT_USERID
zastosowanie 43
USE_TRUSTED_CONTEXT
zastosowanie 42

SQL0965, kod błędu 153
SQL0969, kod błędu 153
SQL1338, kod błędu 32, 153
SQL30020, kod błędu 153
SQL30060, kod błędu 153
SQL30061, kod błędu 153
SQL30073, kod błędu 153
SQL30081N, kod błędu 153
SQL30082, kod błędu 153
SQL5043N, kod błędu 153
SQLCA (obszar komunikacyjny SQL)
  bufory danych 141
  pole SQLCODE 141
SQLCODE
  odwzorowanie 63
  pole w obszarze komunikacyjnym SQL 141
SQLDA (obszar deskryptorów SQL)
  wielkość przydziału 101
SQLSTATE
  kody klasy 63
SRVNAM, obiekt 142
START MVS, komenda systemowa 76
status systemu
  GET SNAPSHOT, komenda 70
statyczny SQL
  wydajność 101
  wyniki przetwarzania 6
STOP MVS, komenda systemowa 76
stowarzyszone bazy danych
  żądanie rozproszone 12
strefy czasowe
  przegląd 33
strojenie
  DB2 Connect 93
  DB2 for OS/390 and z/OS 121
  parametry
    AGENTPRI 117
    DIRCACHE 117
    MAXAGENTS 117
    MAXDARI 117
    NUMDB 117
    RQRIOBLK 117
  wydajność
    baza danych 118
    sieć 119
symboliczna nazwa docelowa 37
  rozróżnianie wielkości liter 32
Sysplex
  informacja o priorytetach 114
  obsługa w programie DB2 Connect 113
  odporność na błędy 114
  parametr 33
  równoważenie obciążenia 114
  uwagi dotyczące platformy zSeries 113
  używanie 114
  wymagania konfiguracyjne 115
System i
  DRDA 9
system wspomagania podejmowania decyzji (DSS) 141
systemowy katalog baz danych
  aktualizowanie 31
  wartości 31

Ś

śledzenie
  dane przesyłane między programem DB2 Connect a serwerem 141
  DRDA
    interpretowanie 141
  informacje w buforze dla śledzenia DRDA 148
  plik wyjściowy 141, 142
  przykładowe pliki wyjściowe 144

T

TCP/IP
  ACCSEC, komenda 142
  bezpieczeństwo
    scenariusze 49
    sprawdzanie 48
  DOMAIN 32
  konfigurowanie
    połączenia z hostem 17
  nazwy hostów 37
  nazwy usług 32
  nazwy zdalnych hostów 32, 37
  numery portów 37
  port ponownej synchronizacji 32
  RESPORT 32
  rozszerzenia RFC-1323
    skalowanie okna 123
  SECCHK, komenda 142
  TCPPORT 32
TCP_KEEPALIVE
  parametr konfiguracyjny systemu operacyjnego 91
testowanie
  aktualizacje na wielu serwerach 56
testowanie porównawcze
  wydajność 95
tokeny
  kody SQLCODE 63
transakcje
  aktualizacje na wielu serwerach 9, 55
  aplikacje rozproszone w architekturze rozszerzonej 58
  DB2 Connect Enterprise Edition 26
  jednostka pracy (UOW) 9
  monitory przetwarzania transakcyjnego 26
  obsługa 58
  przepustowość 95
  rozproszone
    obsługiwane serwery 55
  zatwierdzanie dwufazowe 9
Tuxedo
  DB2 Connect Enterprise Edition 26
tworzenie aplikacji 101
  klient DB2 AD 15
  ODBC 15
typ uwierzytelniania CLIENT
  DB2 Connect, informacje 46
typ uwierzytelniania DATA_ENCRYPT 46
typ uwierzytelniania SERVER 46
typ uwierzytelniania SERVER_ENCRYPT 46
typ uwierzytelniania SERVER_ENCRYPT_AES 46
typy danych
  CHAR 124
  INTEGER
    konwersja danych hosta 123
  konwersja
    wpływ na wydajność 123

typy danych *(kontynuacja)*
nieupakowane dziesiętne 123
upakowane dziesiętne 123
VARCHAR
przegląd 124
zmiennopozycyjne
konwersja danych hosta 123
znak 124
typy danych znakowych 124

U

upakowane dziesiętne, typ danych 123
uprawnienia
wiązanie 51
uruchamianie
monitor poprawności DB2 for z/OS 77
uwagi 169
uwierzytelnianie 37
przegląd 46
sprawdzanie 46
typy
CLIENT 46, 48
DATA_ENCRYPT 46
domyślne 46
KERBEROS 46
SERVER 46
SERVER_ENCRYPT 46

V

VARCHAR, typ danych
opis 124
Visual Explain
kurs 166
VM
DRDA
i DB2 Connect 9
VSE
DRDA 9

W

Wartości w katalogu DCS 33
wartość uwierzytelnienia 31
warunki
korzystanie z publikacji 167
wąskie gardła
transakcje 95
wydajność 95
WebSphere
przegląd 21
WebSphere Federation Server
przegląd 7
WebSphere MQ
DB2 Connect 112
węzły
katalogi
aktualizowanie 31
wartości 32
nazwa
arkusz dostosowania katalogu 37
wartości w katalogu węzłów 32
wartości w systemowym katalogu baz danych 31
wiązanie
nazwy pakietów 51
wiązanie *(kontynuacja)*
pakiety 51
programy narzędziowe i aplikacje 51
uprawnienie
znaczniki parametru z przesunięciem 51
wielkość bloków stronicowania 117
wielkość bloku 117
Windows
Monitor wydajności
monitorowanie aplikacji DB2 69
wydajność
aplikacje 101
DB2 Connect
przegląd 95
rozwiązywanie problemów 121
strojenie 93
zwiększanie szybkości przesyłania 121
dostęp ODBC 99
koncentrator połączeń 111
OS/390 121
sieć
sprzęt 124
wpływ procesora wiersza komend (CLP) 101
z/OS 121
zasoby systemowe 120
zestawianie połączeń 111
wysoka dostępność
DB2 Connect 85

X

X/Open, model rozproszonego przetwarzania transakcyjnego (DTP) 26
XA
menedżery zasobów 26
przykłady koncentratorów 107
zaufane połączenia 41

Z

z/OS
DRDA 9
zalecane działania
przeglądanie, wprowadzanie, zapisywanie 78
zamawianie podręczników do programu DB2 162
zarządzanie połączeniami 105
zasoby systemowe
rywalizacja 120
zatrzymywanie
monitor poprawności DB2 for z/OS 77
zatwierdzanie dwufazowe
port ponownej synchronizacji połączeń TCP/IP 32
włączanie 55
zaufane połączenia 41
przełączanie użytkowników za pośrednictwem interfejsu CLI/ODBC 43
za pośrednictwem interfejsu CLI/ODBC 42
zaufany kontekst
obsługa w programie DB2 Connect 41
za pośrednictwem interfejsu CLI/ODBC 42
zdalna jednostka pracy
charakterystyka 11
przegląd 11
przykład 11
zestaw danych programu startowego (BSDS), parametry
Z/OS i OS/390 32

zestawianie połączeń 105
  porównanie z koncentratorem połączeń 111
  przegląd 105
złożona instrukcja SQL ATOMIC
  nieobsługiwana w programie DB2 Connect 101
złożona instrukcja SQL NOT ATOMIC
  projektowanie aplikacji 101
złożony SQL
  NOT ATOMIC 101
zmiennopozycyjne
  typ danych 123
zrzut pliku śledzenia
  przegląd 138

Ż

żądania bazy danych
  grupowanie w celu poprawy wydajności 101
żądania rozproszone
  kompensacja 12
  obsługa w programie DB2 Connect 12
  przegląd 12
  przezroczystość położenia 12
  stowarzyszone bazy danych 12

SC85-0304-03