



**Ghidul utilizatorului DB2 Connect
Actualizat, noiembrie 2009**



Ghidul utilizatorului DB2 Connect
Actualizat, noiembrie 2009

Notă

Înainte de a folosi aceste informații și produsul la care se referă, aveți grijă să citiți informațiile cu caracter general din Anexa B, "Observații", la pagina 167.

Observație privind ediția

Acest document conține informații care sunt proprietatea IBM. Sunt furnizate pe baza unui acord de licență și sunt protejate de legea referitoare la copyright. Informațiile din această publicație nu includ garanții pentru produs, orice enunț din acest manual trebuind să fie interpretat în acest sens.

Puteți comanda publicațiile IBM online sau prin reprezentantul IBM local.

- Pentru a comanda publicațiile online, mergeți la IBM Publications Center, la www.ibm.com/shop/publications/order
- Pentru a găsi reprezentantul IBM local, mergeți la IBM Directory of Worldwide Contacts, la www.ibm.com/planetwide

Pentru a comanda publicații DB2 la DB2 Marketing and Sales în Statele Unite sau Canada, sunați la 1-800-IBM-4YOU (426-4968).

Când trimiteți informații la IBM, acordați IBM-ului dreptul neexclusiv de a folosi sau distribui informațiile în orice fel considerat corespunzător, fără ca aceasta să implice vreo obligație pentru dumneavoastră.

© Copyright International Business Machines Corporation 1993, 2009.

Cuprins

Despre această carte v

Partea 1. Concepte DB Connect . . . 1

Capitolul 1. DB2 Connect 3

Oferta de produse DB2 Connect 3
Funcțiile livrate în DB2 Connect Versiunea 8 3
Bazele de date gazdă 4
DB2 Connect și instrucțiunile SQL 5
Utilitățile de administrare DB2 Connect 5
InfoSphere Federation Server și DB2 Connect 6

Capitolul 2. Distributed Relational Database Architecture. 7

DRDA și accesul la date 7
DB2 Connect și DRDA 7
Unitatea de lucru la distanță 8
Cereri distribuite 10

Capitolul 3. Scenarii pentru DB2 Connect 11

Acces direct la bazele de date gazdă 11
Accesarea datelor DB2 de pe gazdă System z sau IBM i folosind DB2 Connect Personal Edition 12
Produsele de server DB2 Connect ca servere de conectivitate 14
DB2 Connect și aplicațiile de Web 15
DB2 Connect și IBM WebSphere 16
DB2 Connect ca server de aplicații Java 17
DB2 Connect pe serverul de Web 18
DB2 Connect și serverele de aplicații 18
DB2 Connect și monitoarele de procesare a tranzacțiilor 21

Partea 2. Referințe DB2 Connect . . . 25

Capitolul 4. Actualizarea directoarelor bazei de date 27

Valorile din directorul bazei de date de sistem 27
Valorile directorului de noduri 28
Valorile din directorul DCS 29
Fișă de lucru pentru personalizarea directorului 33
Definirea mai multor intrări pentru aceeași bază de date 33
Tratarea datelor BiDi 34

Capitolul 5. Securitatea DB2 Connect 37

Conexiunile de încredere prin DB2 Connect 37
Crearea și finalizarea unei conexiuni de încredere prin CLI 38
Comutarea utilizatorilor pe o conexiune de încredere prin CLI. 39
Considerente privind autentificarea DB2 Connect 41
Suportul Kerberos 43
Sugestii și indicii privind securitatea z/OS. 43

Tipurile de autentificare suportate cu DB2 Connect 44

Capitolul 6. Legarea aplicațiilor și utilităților (DB2 Connect). 47

Capitolul 7. Actualizările multilocație . . . 51

Activarea actualizărilor multilocație folosind Control Center 52
Testarea actualizărilor multilocație folosind Control Center 52
Actualizarea multilocație și managerul de puncte de sincronizare. 52
Configurarea DB2 Connect cu un manager de tranzacții compatibil XA 53
Suportul DB2 Connect pentru tranzacții cuplate slab 54

Capitolul 8. Mutarea datelor cu DB2 Connect 55

Capitolul 9. Maparea SQLCODE 57

Dezactivarea mapării SQLCODE 57
Ajustarea mapării SQLCODE 57

Capitolul 10. Monitorizarea sistemului de bază de date și DB2 Connect 61

Monitorizarea conexiunilor pentru clienți la distanță. 61
Monitorizarea performanțelor utilizând Windows Performance Monitor 61
Utilizarea comenzilor GET SNAPSHOT 62
Starea aplicației DCS 64
Monitorul de sănătate și alertele. 68
Privire generală asupra monitorului de sănătate DB2 for z/OS 68
Pornirea, oprirea și reîmprospătarea monitorului de sănătate DB2 for z/OS 69
Vizualizarea, lansarea și salvarea acțiunilor recomandate 70
Vizualizarea sumarelor de alerte privind sănătatea 72
Vizualizarea obiectelor de alertă privind sănătatea 74

Partea 3. Disponibilitatea înaltă și DB2 Connect 75

Capitolul 11. Disponibilitatea înaltă și echilibrarea încărcării pentru conectivitatea bazei de date gazdă . . . 77

Capitolul 12. Descrierea și setarea rerutării automate a clientului (DB2 Connect) 79

Capitolul 13. Configurarea re-rutării automate a clientului pentru tehnologia de distribuitor a conexiunii client . . . 81

Partea 4. Ajustarea și DB2 Connect 83

Capitolul 14. Considerente privind performanța DB2 Connect 85

Capitolul 15. Optimizarea accesului ODBC 89

Capitolul 16. Proiectarea aplicației . . . 91

Capitolul 17. Gestionarea conexiunilor 95

Punerea în pool a conexiunilor 95
Concentratorul de conexiuni 97
Punerea conexiunilor în pool și concentratorul de conexiuni 101
Concentratorul de conexiuni este necesar cu WebSphere MQ Transaction Manager și DB2 for z/OS 101

Capitolul 18. Suportul Sysplex pentru serverul DB2 Connect. 103

Considerente pentru exploatarea System z SYSPLEX 103
Exploatarea DB2 Sysplex 104
Cerințele de configurare pentru Sysplex 105

Capitolul 19. Suportul Sysplex pentru client 107

Echilibrarea încărcării de lucru la nivel de tranzacție (pe partea client) 107
 Configurarea echilibrării încărcării de lucru la nivel de tranzacție (pe partea client) 108
 Rerutarea automată a clientului (pe partea client) 110
 Configurarea rerutării automate a clientului (pe partea client) 111
Configurarea afinităților clientului 114
Limitări la folosirea suportului de Sysplex pentru client 117

Capitolul 20. Ajustarea DB2 Connect 119

Ajustarea bazei de date gazdă 120
Considerente ale ajustării rețelei 121
Conflictul pentru resursele de sistem 122
Depanarea performanței DB2 Connect 122
Ajustarea DB2 for z/OS 123
Creșterea ratelor de transfer de date DB2 Connect 123
Blocul de interogare suplimentar 123
Scalarea ferestrei RFC-1323 124
Conversia datelor gazdă 125
Tipurile de date pentru datele caracter 125
Hardware-ul de rețea 126

Capitolul 21. Ajustarea performanței aplicației CLI/ODBC. 127

Partea 5. Depanarea 129

Capitolul 22. Depanarea DB2 Connect 131

Strângerea informațiilor relevante 131
Conectarea inițială nu se realizează cu succes 131
Au apărut probleme după conectarea inițială. 132
Uneltele de diagnoză 133

Capitolul 23. Urmăririle DB2 în DB2 Connect. 135

Obținerea unei urme DB2 folosind db2trc 135
Realizarea unui dump pentru un fișier de urmărire DB2 136
Formatarea unui fișier de urmărire DB2 136

Capitolul 24. Fișierele de urmărire DRDA. 139

Utilitarul de urmărire 139
Ieșirea urmăririi 140
Analizarea fișierului cu ieșirea urmăririi 140
Exemple de fișier cu ieșire a urmăririi 142
Informațiile ulterioare de buffer pentru urmărirea DRDA 146

Partea 6. Mesajele 149

Capitolul 25. Probleme obișnuite DB2 Connect. 151

Partea 7. Anexe. 155

Anexa A. Privire generală asupra informațiilor tehnice DB2 157

Biblioteca tehnică DB2 în format tipărit sau PDF 157
Comandarea cărților DB2 tipărite 160
Afișarea ajutorului pentru starea SQL din linia de comandă a procesorului 161
Accesarea diferitelor versiuni de Centru de informare DB2 161
Afișarea subiectelor în limba preferată în Centrul de informare DB2 161
Actualizarea Centrului de informare DB2 instalat pe calculatorul dumneavoastră sau pe un server din intranet 162
Actualizarea manuală a Centrului de informare DB2 instalat pe calculatorul dumneavoastră sau pe un server din intranet 163
Îndrumările DB2. 165
Informații privind depanarea DB2. 165
Termenii și condițiile 166

Anexa B. Observații. 167

Index 171

Despre această carte

Ghidul utilizatorului DB2 Connect oferă informațiile necesare pentru a vă familiariza cu produsul DB2 Connect și folosirea acestuia. Sunt prezentate concepte privind DB2 Connect cu scenarii tipice, ce arată relațiile dintre DB2 Connect și alte părți componente ale mediului de lucru în rețea. Sunt discutate considerentele pentru directoarele bazei de date, securitatea între sisteme, actualizările multilocație, mutarea datelor și monitorizarea DB2 Connect. Se arată cum suportă DB2 Connect disponibilitatea înaltă în mediul dumneavoastră de lucru în rețea. Sunt prezentate noțiuni introductive pentru asigurarea performanței de către DB2 Connect și în rețea, precum și unele subiecte referitoare la depanarea posibilelor probleme.

Cui îi este destinată această carte?

În această carte (sau porțiuni ale ei), pot găsi informații de interes administratorii de sistem, administratorii de bază de date și specialiștii în comunicațiile sistemelor.

Partea 1. Concepte DB Connect

Capitolul 1. DB2 Connect

DB2 Connect oferă conectivitate rapidă și robustă la bazele de date de pe sisteme mainframe IBM® pentru e-business și alte aplicații care rulează sub sistemele de operare Linux®, UNIX® și Windows®.

DB2 Connect Personal Edition furnizează conectivitate directă pentru servere System z și IBM Power Systems, în timp ce produsele server DB2 Connect furnizează conectivitate indirectă care permite clienților să acceseze servere System z și IBM Power Systems prin gateway-ul DB2 Connect. O varietate de produse server DB2 Connect furnizează împachetare unică și soluții de licențiere care vă permit să selectați un produs care este potrivit pentru mediul dumneavoastră.

Oferta de produse DB2 Connect

DB2 Connect oferă câteva soluții de conectare, cum ar fi DB2 Connect Personal Edition și mai multe produse server DB2 Connect.

- DB2 Connect Enterprise Edition
- DB2 Connect Application Server Edition
- DB2 Connect Unlimited Edition for System z
- DB2 Connect Unlimited Edition for System i

Pentru informații detaliate despre ofertele produsului DB2 Connect, vedeți www.ibm.com/software/data/db2/db2connect/

Funcțiile livrate în DB2 Connect Versiunea 8

Această secțiune furnizează un rezumat al îmbunătățirilor introduse în DB2 Connect Versiunea 8. Pentru a găsi lista de modificări introduse în DB2 Versiunea 9 care afectează funcționalitatea DB2 Connect, vedeți următoarele subiecte:

- Sumarul pachetelor de corecții DB2 Connect Versiunea 9.5
- Sumarul pachetelor de corecții DB2 Connect Versiunea 9.1

Funcțiile livrate în DB2 Connect Versiunea 8 Ediția 2

DB2 Connect Versiunea 8.2 include următoarele îmbunătățiri:

- Rerutarea automată a clientului

Dacă este pierdută o conexiune TCP/IP la un server sau server DB2 Connect, clientul va încerca automat să restabilească conexiunea dacă există un server alternativ. Serverul alternativ este specificat în instanța de server, locația sa fiind trimisă clientului în timpul conexiunii.

- Criptarea datelor

Comunicațiile client/server asigură acum criptarea datelor de utilizator în timp ce circulă prin rețea.

Funcțiile livrate în DB2 Connect Versiunea 8 Ediția 1 (inclusiv toate pachetele de corecții și nivelurile de modificare)

DB2 Connect Versiunea 8.1 include următoarele îmbunătățiri:

- Suport pentru instrucțiuni SQL mai lungi (până la 2 MB)

Prin aplicațiile CLI și JDBC pot circula instrucțiuni SQL de până la 2 MB. Interfața încorporată rămâne însă la limita de 64 K.

- Informații de diagnoză care identifică originea unei instrucțiuni SQL
Furnizează abilitatea de a determina care program aplicație a emis o anumită instrucțiune în cache-ul de instrucțiuni SQL dinamic DB2 for z/OS.
- Matrice de intrări pe coloane
Permite aplicațiilor să ofere mai multe seturi de parametri pentru o singură instrucțiune SQL.
- Monitorizarea timpului de rețea
Acum se folosesc noi elemente de monitorizare, pentru a obține o și mai bună imagine a activității bazei de date și a traficului din rețea la nivel de bază de date sau de aplicație.
- Suport cursor care suportă defilare DB2 CLI dinamic
Cursoarele cu defilare dinamică sunt suportate acum în DB2 CLI când sunt accesate servere DB2 Universal Database (UDB) for z/OS Versiunea 8.1 sau ulterioare.
- Suport eWLM
Oferă capacitatea de a monitoriza cap la cap unitățile de lucru prin grupurile middleware pentru a determina gâtuirile.
- Îmbunătățiri ale comenzii DB2 ping
Comanda DB2 ping suportă acum specificația unei dimensiuni pachet cerere și răspuns.

Notă: DB2 Connect nu suportă comanda PING când este lansată către gazdă dintr-un client Versiunea 7 printr-un gateway Versiunea 9.

Bazele de date gazdă

Termenul *bază de date* este folosit în acest document pentru a descrie un sistem relațional de gestionare a bazei de date (RDBMS). Alte sisteme cu care comunică DB2 Connect ar putea folosi termenul bază de date pentru a descrie un concept puțin diferit. Termenul DB2 Connect bază de date se mai poate referi la:

System z

DB2 for z/OS. Un subsistem DB2 for z/OS este identificat de numele său de locație (LOCATION NAME). Valoarea LOCATION NAME poate fi determinată prin logarea pe TSO și lansarea următoarei interogări SQL folosind una dintre următoarele unelte de interogare disponibile:

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

LOCATION NAME (nume locație) este de asemenea definit în BSDS (Boot Strap Data Set), cât și în mesajul DSNL004I (LOCATION=locație), care este scris când pornește DDF (Distributed Data Facility). LOCATION NAME suportă până la 8 nume alias de locații, permițând aplicațiilor să folosească diferite nume dbalias pentru a accesa un server Versiunea 8 z/OS. Folosiți comanda z/OS -display ddf pentru a obține numele locației server, numele de domeniu, adresă IP și portDB2.

VSE DB2 for VSE rulând într-o partiție bază de date identificată prin DBNAME

VM DB2 for VM rulând într-o mașină virtuală CMS identificată prin DBNAME

Servere IBM Power Systems

DB2 for IBM i, o parte integrată în sistemul de operare IBM i. Doar o bază de date poate exista pe un server IBM Power Systems dacă sistemul nu este configurat să folosească pool-uri de stocare auxiliare independente.

DB2 Connect și instrucțiunile SQL

DB2 Connect înaintează instrucțiunile SQL lansate de programe de aplicație la serverele de bază de date mainframe IBM.

DB2 Connect poate înainta aproape orice instrucțiune validă SQL, precum și API-urile suportate DB2 (interfețe de programare a aplicațiilor):

- JDBC
- SQLJ
- ADO.NET
- OLE DB
- ODBC
- Perl
- PHP
- pureQuery
- Python
- Ruby
- DB2 CLI
- SQL încorporat

Suportul SQL încorporat

Există două tipuri de procesare SQL încorporat: SQL static și SQL dinamic. SQL Static minimizează timpul necesar pentru a executa o instrucțiune SQL prin procesarea înainte. SQL dinamic este procesat când instrucțiunea SQL este lansată la serverul de bază mainframe IBM. Codul SQL dinamic este mai flexibil, dar poate fi mai lent. Decizia de a folosi SQL static sau dinamic este luată de programatorul aplicației. Ambele tipuri sunt suportate de DB2 Connect.

Serverele de bază de date mainframe IBM diferite implementează SQL în mod diferit. DB2 Connect suportă complet IBM SQL comun, precum și implementările DB2 for z/OS, DB2 Server for VM and VSE (numit înainte SQL/DS) și DB2 for IBM i de SQL. IBM SQL este puternic recomandat pentru menținerea independenței bazei de date.

Utilitarele de administrare DB2 Connect

Important: Control Center și componentele sale asociate au fost depreciate în Versiunea 9.7 și s-ar putea să fie înlăturate în ediția viitoare. Pentru informații suplimentare, vedeți subiectul “Uneltele Control Center și DB2 Administration Server (DAS) au fost depreciate” în cartea *Ce este nou pentru Versiunea 9.7*.

Sunt disponibile următoarele utilitare pentru a-l ajuta pe un administrator DB2 Connect:

- Command Line Processor (CLP) vă permite să lansați instrucțiuni SQL pentru o bază de date a unui server de baze de date mainframe IBM. Dirijează instrucțiunile SQL la baza de date pe care o specificați.
- DB2 Command Center furnizează o interfață grafică pentru Command Line Processor (CLP).
- Utilitarele de import și export vă permit să încărcați, importați și exportați date în și dintr-un fișier pe o stație de lucru și o bază de date a unui server de baze de date mainframe IBM. Aceste fișiere pot fi folosite pentru importul datelor în baze de date, foi de calcul tabelar și alte aplicații ce rulează pe stația dumneavoastră de lucru.

- Dacă rulați un produs server DB2 Connect, puteți folosi Event Viewer și Performance Monitor. Folosind Event Viewer, puteți vizualiza evenimentele excepție înregistrate de DB2 Connect. Folosind Performance Monitor, puteți monitoriza și gestiona performanțele serverelor DB2 Connect locale sau de la distanță
- DB2 Control Center vă permite să administrați și să monitorizați toate aspectele serverelor DB2 Connect. Permite de asemenea administratorilor să lucreze cu obiecte bază de date DB2 for z/OS, cum ar fi tabele, vizualizări, pool-uri de buffer-e și fire de execuție.
- Utilitarul monitor de sistem bază de date lasă administratorul de sistem să monitorizeze conexiunile sistemului. Această funcție este disponibilă doar când DB2 Connect acționează ca un server. Acest utilitar de asemenea ajută administratorul de sistem să determine sursa unei erori. Administratorul sistemului poate corela aplicații client cu joburile corespunzătoare care rulează pe serverul de bază de date mainframe IBM.

Notă: În edițiile anterioare, DB2 Graphical Administration Tools, cum ar fi Control Center (centrul de control), erau suportate pe toate platformele. De la Versiunea 9, DB2 Graphical Administration Tools sunt suportate doar pe Windows x86, Windows x64 (AMD64/EM64T), Linux pe x86 și Linux pe AMD64/EM64T. Pentru toate platformele, puteți folosi DB2 CLP în scopuri de administrare.

InfoSphere Federation Server și DB2 Connect

InfoSphere Federation Server este o ofertă de produs separat care furnizează acces la o integrare a datelor de-a lungul surselor de date multivendor, în timp ce DB2 Connect vă permite să folosiți volumele mari de date localizate pe serverele gazdă și midrange existente.

InfoSphere Federation Server ajută la integrarea informațiilor permițând o colecție de surse de date pentru a fi vizualizate și manipulate ca și cum ar fi o singură sursă. Acesta face complet transparent accesul la sursa de date pentru aplicația apelantă. InfoSphere Federation Server lucrează în conjuncție cu produsele server DB2 Connect. InfoSphere Federation Server furnizează acces de citire și scriere nativ la familia de produse DB2, Informix, Oracle, Sybase, Teradata și baze de date Microsoft® SQL Server. InfoSphere Federation Server furnizează de asemenea acces de citire la surse de date nonrelaționale și surse de date pentru științele naturii, cum ar fi Documentum, IBM Lotus Extended Search, fișierele structurate tabelă și XML. Îl puteți utiliza pentru a formula interogări pe datele dintr-un sistem federalizat.

Capitolul 2. Distributed Relational Database Architecture

Distributed Relational Database Architecture (DRDA) este un set de protocoale care permit să lucreze împreună mai multe sisteme de bază de date, atât IBM, cât și non-IBM, precum și programe de aplicație. Poate fi conectată orice combinație de produse de gestionare a bazelor de date relaționale care folosesc DRDA, pentru a forma un sistem de gestionare a bazei de date relaționale distribuite. DRDA coordonează comunicațiile între sisteme definind ce trebuie schimbat și cum trebuie schimbat.

Unitate de lucru

O *unitate de lucru (UOW)* este o singură tranzacție logică. Constă dintr-o secvență de instrucțiuni SQL în care fie toate operațiile sunt realizate cu succes, fie secvența ca un tot este considerată eșuată.

Unitate de lucru distribuită

O *unitate de lucru distribuită (DUOW)*, numită și actualizare multilocație, implică mai multe servere de bază de date într-o unitate de lucru. O DUOW are următoarele caracteristici:

- Mai mult de un server de gestionare a bazei de date este actualizat de unitatea de lucru.
- Aplicația direcționează distribuția lucrului și inițializează comiterea.
- Ar putea fi mai multe cereri pentru unitatea de lucru.
- Există un server de gestionare a bazei de date pe cerere.
- Comiterea este coordonată de-a lungul serverelor baze de date multiple.

DRDA și accesul la date

Chiar dacă DRDA definește protocoale de comunicație bază de date, nu definește interfețele de programare, sau API-urile, care ar trebui folosite de programatorii de aplicații. În general, DRDA poate fi folosit de un program aplicație pentru a pasa orice cerere pe care un server destinație DRDA o poate executa. Toate serverele DRDA disponibile în ziua de azi pot executa cereri SQL înaintate de un program aplicație prin DB2 Connect.

IBM furnizează programatorilor de aplicații unelte pentru a genera cereri SQL pentru sistemele de operare Windows, UNIX și Linux. Aceste unelte fac parte din clientul DB2. Managerul de bază de date DB2 suportă mai multe interfețe de programare: ADO.NET, JDBC, SQLJ, PHP, Perl DBI, SQL încorporat, DB2 Call Level Interface (CLI) și OLE DB. Aceste API-uri pot fi folosite de programatori pentru a construi aplicații în diverse limbaje de programare.

DB2 Connect și DRDA

DB2 Connect implementează arhitectura DRDA pentru a reduce costul și complexitatea accesării datelor stocate în DB2 for IBM i, DB2 for IBM Power Systems, DB2 for z/OS, DB2 Server for VM and VSE și alte servere de bază de date compatibile DRDA. Exploatând complet arhitectura DRDA, DB2 Connect oferă o soluție performantă, de cost scăzut, cu caracteristici de gestionare a sistemului pe care le cer clienții.

În tehnologia DRDA, un *solicitant de aplicație (AR - application requester)* este codul care manipulează terminarea aplicației a unei conexiuni distribuite. AR este aplicația care cere

date. DB2 Connect se poartă ca un solicitant de aplicație în numele programelor aplicație care pot fi locale stației de lucru DB2 Connect sau pe un client separat la distanță pentru DB2 Connect.

Un *server de aplicații (AS - application server)* este codul care manipulează capătul de bază de date al conexiunii.

DRDA suportă de asemenea conexiuni multinivel între un solicitant de aplicație și un server. În această topologie, serverul la care se conectează un solicitant de aplicație este un server de aplicații, iar orice server din aval se numește server de bază de date (DS - database server), deoarece nu interacționează direct cu solicitantul de aplicație. În plus, pentru a evidenția rolul său, care nu este nici acela de sistem de la care pleacă o solicitare de bază, nici acela de sistem care execută funcția de bază de date cerută, fiecare server de aplicații sau server de bază de date dintre un solicitant de aplicație și serverul final de bază de date este numit de asemenea server intermediar. Folosirea serverelor bază de date și serverelor intermediare este suportată de DB2 Connect.

Figura 1 arată fluxul datelor dintre stația de lucru DB2 Connect și serverul mainframe IBM în cazul în care există doar clienți locali.

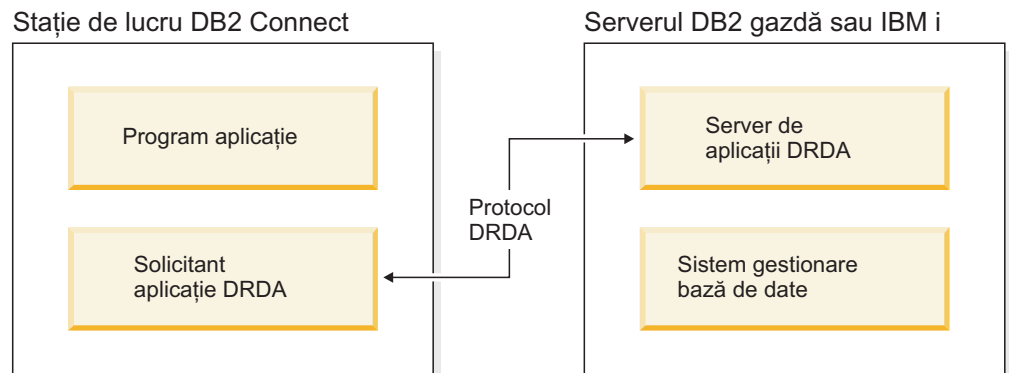


Figura 1. Fluxul de date între un server DB2 Connect și un server mainframe IBM

Pentru a implementa conexiunile dintre sisteme gestiune bază de date server DRDA și IBM Data Server-uri, DRDA folosește următoarele arhitecturi:

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

Aceste arhitecturi sunt folosite ca blocuri de construire. Fluxurile de date care curg prin rețea sunt specificate de arhitectura DRDA care documentează un protocol flux de date care suportă acces bază de date relațional distribuit.

O cerere este rulată la destinația corectă prin intermediul directoarelor care conțin diverse tipuri de informații de comunicare și numele bazei de date server DRDA care este accesată.

Unitatea de lucru la distanță

O *unitate de lucru la distanță* permite unui utilizator sau unei aplicații program să citească sau să actualizeze datele dintr-o locație per unitatea de lucru. Oferă suport pentru accesul la o singură bază de date dintr-o unitate de lucru. Un program de aplicație poate actualiza mai multe baze de date la distanță, dar poate accesa o singură bază de date dintr-o unitate de lucru.

Unitatea de lucru de la distanță are următoarele caracteristici:

- Sunt suportate mai multe cereri (instrucțiuni SQL) per unitatea de lucru.
- Sunt suportate mai multe cursoare per unitatea de lucru.
- Fiecare unitate de lucru poate actualiza o singură bază de date.
- Programul de aplicație fie comite, fie derulează înapoi unitatea de lucru. În anumite circumstanțe de eroare, serverul de bază de date sau DB2 Connect ar putea întoarce unitatea de lucru.

De exemplu, Figura 2 arată un client bază de date care rulează o aplicație transfer de fonduri care accesează o bază de date care conține tabele cont de economii, precum o planificare a taxei de tranzacție. Aplicația trebuie:

- Să accepte suma de transferat de la interfața de utilizator.
- Să scadă suma din contul de economii și să determine noul sold.
- Să citească planificarea taxei pentru a determina taxa de tranzacție pentru un cont de economii cu un anumit sold.
- Să scadă taxa de tranzacție din contul de economii.
- Să adauge suma transferului în contul curent.
- Să comită tranzacția (unitatea de lucru).

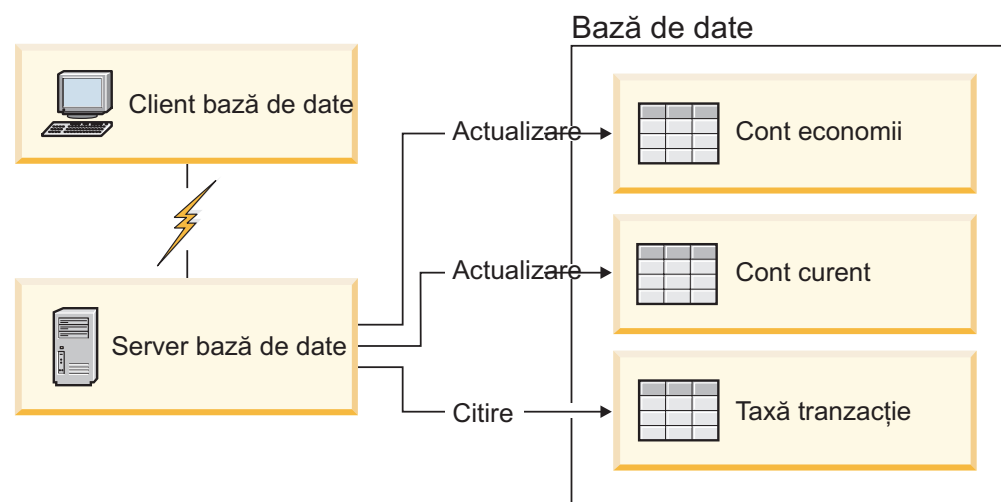


Figura 2. Folosirea unei singure baze de date într-o tranzacție

Pentru a seta o asemenea aplicație, dumneavoastră trebuie:

1. Să creați tabelele pentru contul de economii, verificând planificarea taxelor de cont și tranzacție din aceeași bază de date.
2. În cazul distanței fizice, să setați serverul bazei de date pentru folosirea protocolului de comunicație corespunzător.
3. În cazul distanței fizice, să catalogați nodul și baza de date pentru a identifica baza de date de pe serverul bazei de date.
4. Să precompilați programul de aplicație pentru a specifica o conexiune de tip 1; cu alte cuvinte, să specificați CONNECT(1) în comanda PREP.

Cereri distribuite

O *cerere distribuită* este o funcție de bază de date distribuită care permite aplicațiilor și utilizatorilor să lanseze instrucțiuni SQL care referă două sau mai multe DBMS-uri sau baze de date într-o singură instrucțiune. De exemplu, o uniune între tabele din două subsisteme DB2 for z/OS diferite.

DB2 Connect furnizează suport pentru cereri distribuite de-a lungul bazelor de date și DBMS-uri. De exemplu, puteți realiza o operație de UNION între o tabelă DB2 și o vizualizare Oracle. DBMS-urile suportate includ membri DB2 Family (cum ar fi DB2 Database for Linux, UNIX, and Windows, DB2 for z/OS și DB2 for i) și Oracle. Când se folosește DB2 Connect împreună cu InfoSphere Federation Server, este disponibil suport pentru mai mulți furnizori.

Cererea distribuită asigură *transparența locației* pentru obiectele bazei de date. Dacă informațiile (din tabele și vizualizări) sunt mutate, referințele la acele informații (numite *porecle*) pot fi actualizate fără a fi necesară modificarea aplicațiilor care cer informațiile. Cererea distribuită furnizează de asemenea *compensare* pentru DBMS-uri care nu suportă toate dialectele DB2 SQL sau anumite capacități de optimizare. Operațiile care nu pot fi realizate sub un asemenea DBMS (cum ar fi un SQL recursiv) sunt rulate sub DB2 Connect.

Cererea distribuită funcționează într-o manieră *semiautonomă*. De exemplu, interogările DB2 care conțin referințe la obiecte Oracle pot fi lansate în timp ce aplicațiile Oracle accesează același server. Cererile distribuite nu monopolizează și nici nu restricționează accesul (în afara constrângerilor de integritate și de blocare) la obiectele Oracle sau alte obiecte DBMS.

Implementarea funcției de cerere distribuită conține o instanță DB2 Connect, o bază de date care va servi ca baza de date federalizată și una sau mai multe surse de date. *Baza de date federalizată* conține intrări de catalog ce identifică sursele de date și caracteristicile lor. O *sursă de date* constă dintr-un DBMS și datele respective. Aplicațiile se conectează la baza de date federalizată precum orice copie a altei baze de date DB2. Baza de date federalizată DB2 Connect nu este licențiată pentru gestionarea datelor utilizatorilor. Singurul scop este de a conține informații despre sursele de date.

După ce este setat un sistem federalizat, informațiile din sursele de date pot fi accesate ca și cum ar fi într-o bază de date mai mare. Utilizatorii și aplicațiile trimit interogări la o bază de date federalizată, care apoi extrage date din sistemele familie DB2 și Oracle sunt necesare. Utilizatorii și aplicațiile specifică porecle în interogare; aceste porecle furnizează referințe la tabele și vizualizări localizate în sursele datelor. Din perspectiva unui utilizator final, porecele sunt similare cu aliasurile.

Mulți factori pot afecta performanța cererilor distribuite. Cel mai important este să vă asigurați că în catalogul global al bazei de date federalizate sunt memorate informații corecte și actualizate despre sursele datelor și obiectele lor. Aceste informații sunt folosite de optimizatorul DB2 și pot afecta deciziile pentru a coborî operațiile pentru evaluare la surse de date.

Capitolul 3. Scenarii pentru DB2 Connect

DB2 Connect poate oferi o varietate de soluții pentru necesitățile de acces la baza de date mainframe IBM. Acest subiect prezintă mai multe scenarii care s-ar putea aplica la mediul sau necesitățile dumneavoastră particulare.

Acces direct la bazele de date gazdă

Caracteristica de bază a lui DB2 Connect furnizează o conexiune directă la o bază de date gazdă de la aplicații desktop care rulează pe stațiile dumneavoastră de lucru. Cea mai simplă soluție pentru a furniza aceasta este IBM Data Server Driver Package cu licență DB2 Connect.

Fiecare stație de lucru care are DB2 Connect Personal Edition instalat poate stabili o conexiune TCP/IP directă la serverele DB2 for z/OS, DB2 for IBM i și DB2 Database for Linux, UNIX, and Windows. În plus, aplicațiile se pot conecta și pot actualiza mai multe baze de date din familia DB2 din aceeași tranzacție cu integritate completă a datelor furnizată de protocolul de comitere în două faze.

Figura 3 la pagina 12 arată o conexiune directă la un server de bază de date mainframe IBM de la o stație de lucru cu DB2 Connect Personal Edition instalat.

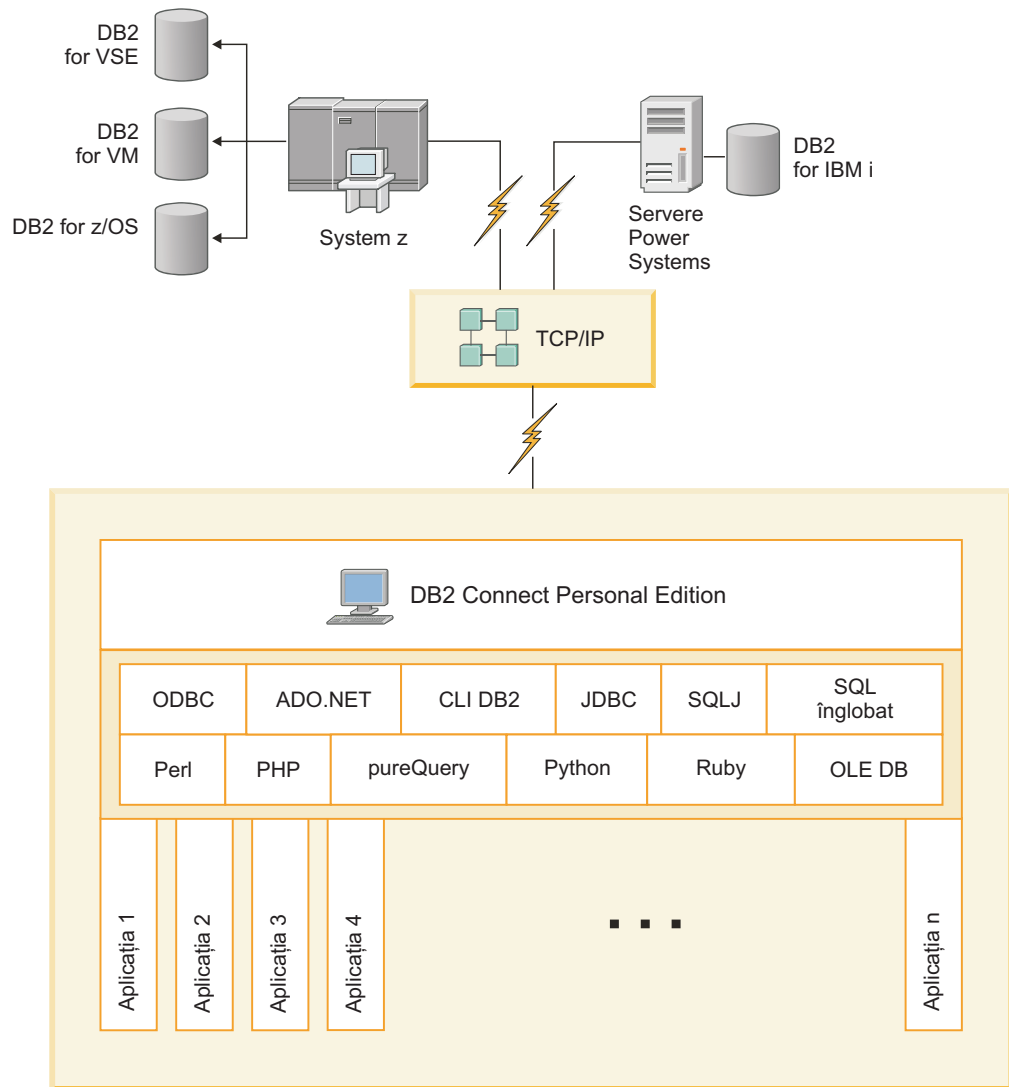


Figura 3. Conexiunea directă între DB2 Connect și un server de bază de date mainframe IBM

Notă:

1. Nu trebuie să aveți DB2 instalat pe stația de lucru DB2 Connect Personal Edition. Dacă vreți un sistem complet de gestionare a bazelor de date relaționale pe stația de lucru DB2 Connect Personal Edition, comandați DB2.
2. Toată funcționalitatea IBM Data Server Client este disponibilă cu DB2 Connect Personal Edition.
3. Dacă o conexiune la un server bază de date DB2 for z/OS cu exploatare Sysplex activată este pierdut, clientul va încerca automat să restabilească conexiunea.

Accesarea datelor DB2 de pe gazdă System z sau IBM i folosind DB2 Connect Personal Edition

Conexiunea directă, fără servere intermediare, este o configurație foarte comodă și utilă. Aceasta mai ales în situațiile în care serverul de bază de date mainframe IBM suportă conectivitatea TCP/IP. În astfel de situații, fiecare stație de lucru DB2 Connect stabilește o conexiune directă cu serverul de bază de date mainframe IBM.

Conectivitatea TCP/IP necesită ca baza de date mainframe IBM să suporte TCP/IP. Următoarele versiuni suportă conexiuni TCP/IP native:

- DB2 for z/OS Versiunea 7.1 sau mai recentă
- DB2 for IBM i Versiunea 5.1 sau mai recentă, și
- DB2 Server for VM and VSE Versiunea 7 sau mai recentă

Pentru a vă conecta la un server de bază de date mainframe IBM, aveți nevoie de o licență DB2 Connect care poate fi adăugată la un IBM Data Server.

Figura 4 la pagina 14 prezintă o stație de lucru pe care este instalat DB2 Connect Personal Edition, conectată direct la un server de bază de date mainframe IBM.

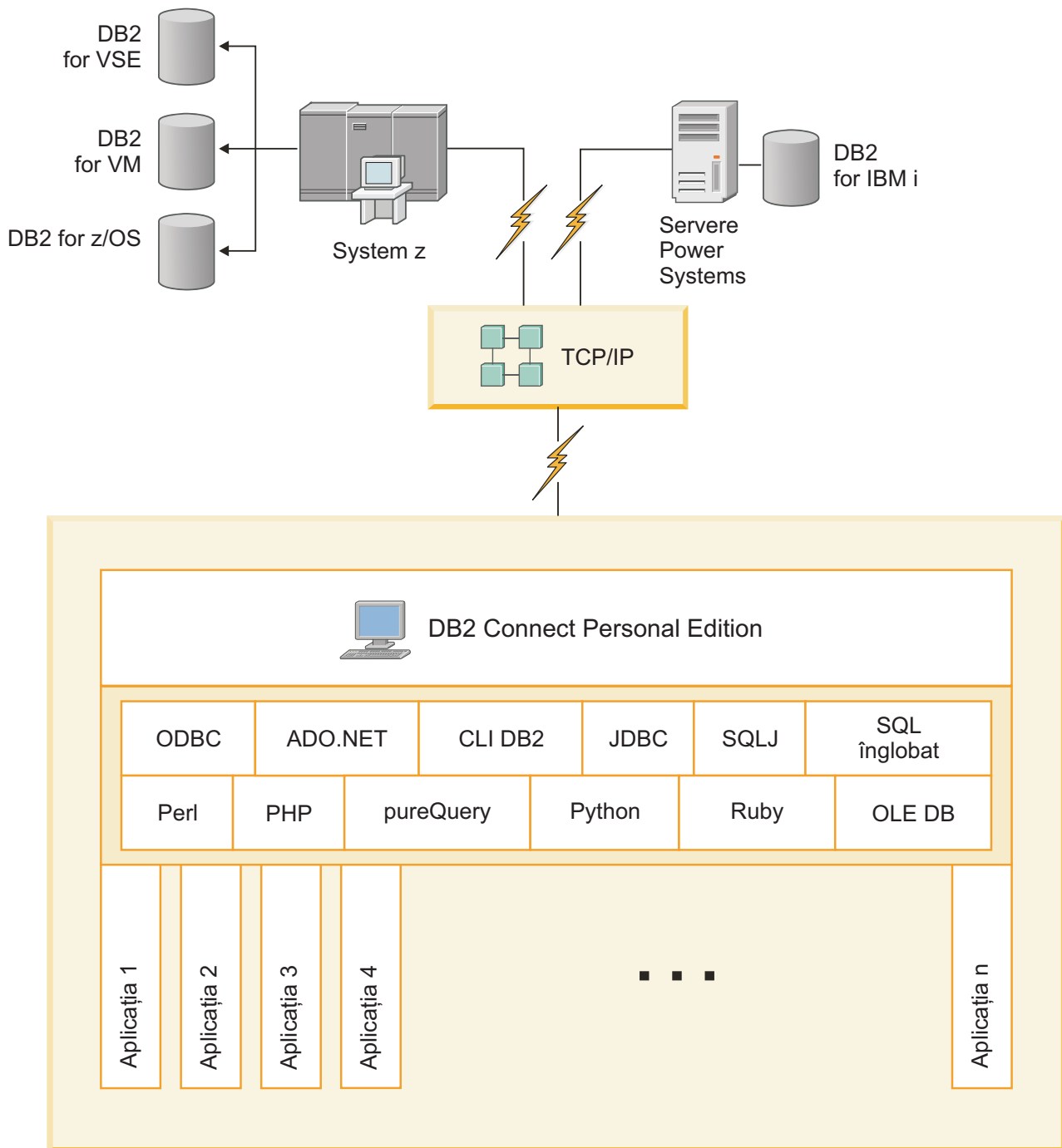


Figura 4. Conexiunea directă între DB2 Connect și un server de bază de date mainframe IBM

Produsele de server DB2 Connect ca servere de conectivitate

Un server DB2 Connect permite conectarea mai multor clienți la datele mainframe-ului IBM, putând reduce semnificativ efortul necesar pentru stabilirea și menținerea accesului la datele de întreprindere. Figura 5 la pagina 15 ilustrează soluția IBM pentru mediile de lucru în care doriți ca un client DB2 să realizeze o conexiune indirectă la un server de bază de date mainframe IBM printr-un produs server DB2 Connect, cum ar fi DB2 Connect Enterprise Edition.

Notă: Conexiunile indirecte sunt suportate cu clienții DB2 sau JCC care rulează pe Linux, UNIX sau Windows. Încercarea de a vă conecta la un server de bază de date mainframe IBM printr-un produs server DB2 Connect folosind orice alt client duce la o eroare SQL1334.

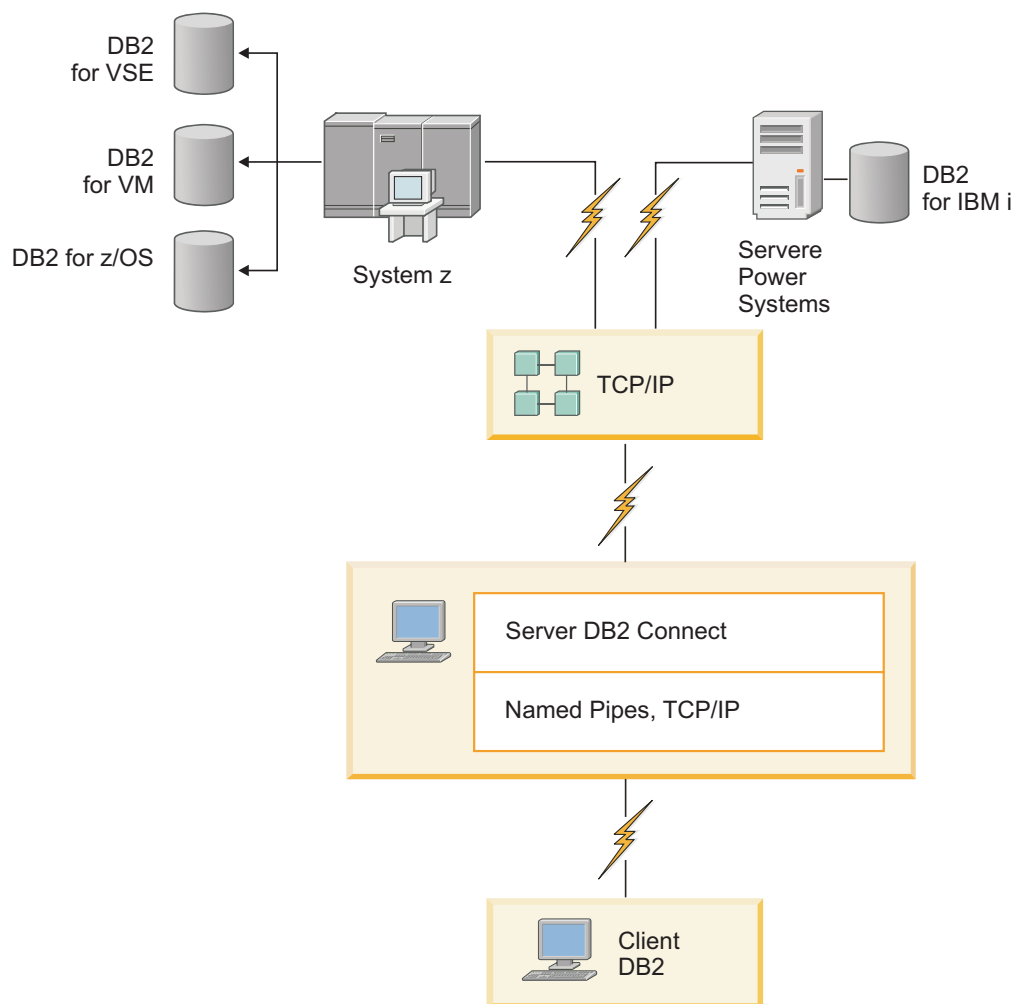


Figura 5. DB2 Connect Enterprise Edition

Dacă este pierdută o conexiune TCP/IP la serverul DB2 Connect, clientul va încerca automat restabilirea conexiunii. Mai întâi clientul încercă restabilirea conexiunii cu serverul original. Dacă nu este restabilită conexiunea, clientul va fi preluat de un server alternativ DB2 Connect. (Serverul alternativ este specificat în instanța de server, locația sa fiind returnată clientului în timpul conexiunii.) În cazul în care conexiunea cu serverul alternativ nu este restabilită, clientul încercă restabilirea conexiunii cu serverul original. Clientul va continua să încerce restabilirea conexiunii, comutând între serverul original și serverul alternativ, până când conexiunea este restabilită sau numărul de încercări depășește valoarea de timeout.

DB2 Connect și aplicațiile de Web

Browser-ul de Web devine rapid o interfață standard pentru orice, de la cataloagele online la aplicațiile de intranet. Pentru aplicații web simple, ar putea fi suficient doar un server web. Pentru aplicații avansate care necesită acces la baze de date și procesare tranzacții, IBM oferă soluții care folosesc DB2 Connect pentru a gestiona numere foarte mari de tranzacții simultane pe web.

Avantaje și limitări ale programării CGI tradiționale

Aplicațiile e-business prin World Wide Web folosesc de obicei Common Gateway Interface (CGI) pentru a permite utilizatorilor să interogheze baze de date back-end. De asemenea, multe companii folosesc intern aplicațiile web, acestea având de obicei o bază de date în fundal.

Utilizatorii completează formularele pe o pagină web și apoi acestea sunt trimise prin CGI aplicațiilor sau scripturilor de pe serverul de Web. Scriptul, la rândul său, folosește un API de bază de date furnizat pentru a lansa interogări SQL într-o bază de date gazdă. Același script poate apoi construi o pagină web (HTML) cu rezultatele interogării și le trimite înapoi, pentru a fi afișate de browser-ul de Web al utilizatorului. Un exemplu este un catalog online unde utilizatorul poate interoga disponibilitatea și prețul curent al anumitor bunuri și servicii.

Aplicațiile CGI pot fi simplu de proiectat și ușor de întreținut. De când standardul CGI a devenit independent de sistemul de operare și de limbaj, este disponibil pe aproape toate platformele. Programele CGI pot fi scrise în C++ sau într-un limbaj de scriptare cum ar fi Perl sau PHP.

În timp ce CGI ar putea părea o soluție ideală pentru aplicații bazate pe web, are dezavantaje semnificative. Mediul de programare pentru CGI nu este la fel de sofisticat ca alte API-uri. În plus, scalabilitatea poate deveni o problemă cu operațiile de e-commerce pe scară largă. De fiecare dată când este invocată o aplicație CGI, este creat un nou proces pe serverul de Web. Fiecare proces trebuie să-și realizeze propria conexiune la baza de date și să-și lanseze propria interogare. În medii cu un număr mare de tranzacții, această limitare poate crea probleme de performanță semnificative.

Puteți folosi DB2 Connect cu un server web pentru a crea aplicații e-commerce robuste și de volum mare. DB2 Connect furnizează mai multe soluții care îmbunătățesc performanța aplicațiilor bazate pe web. Procedurile memorate permit DB2 Connect utilizatorilor să reducă numărul de interogări trimise la baza de date.

Punerea conexiunilor în pool reduce frecvența conectării și deconectării la și de la o bază de date.

Folosirea PHP ca un modul de server Web sau ca plug-in

Deși PHP poate fi folosit pentru programarea CGI, este folosit în mod obișnuit ca un modul de server Web sau ca un plug-in. Într-un server de Web multi-proces cum este Apache, driverul IBM DB2 pentru PHP poate fi folosit pentru a domoli problema de scalabilitate. Într-un server de Web multi-proces, un pool de procese este refolosit pentru a servi cererile serverului de Web. Pentru a nu mai fi nevoie să se construiască o conexiune bază de date pentru fiecare cerere web poate fi creată o conexiune persistentă. În acest mediu, o conexiune persistentă poate exista mai mult decât domeniul unui singur script PHP. Conexiunea va fi reutilizată, dacă este necesară o conexiune identică pentru o cerere de Web ulterioară.

DB2 Connect și IBM WebSphere

IBM WebSphere furnizează o soluție e-business mai completă decât este posibil cu unele tradiționale de scriptare, cum ar fi PHP. Serverele de aplicații WebSphere nu furnizează doar posibilitățile de scriptare ale PHP, dar de asemenea vă permit servicii complexe high-end prin web, folosind servleturi, Active Server Pages și Enterprise JavaBeans™ și includ suport pentru tehnologii bazate pe web, cum ar fi Java™, TCP/IP, HTTP, HTTPS, HTML, DHTML, XML, MIME, SMTP, IIOP și X.509, printre altele. Cu WebSphere puteți să:

- Exploatați standarde industriale pentru a accelera dezvoltarea și maximizarea interoperabilității.

- Includeți cadre de lucru pentru aplicație și tehnologii de unele parte terță
- Analizați performanța și utilizarea conținutului sitului Web
- Scalați cu ușurință situl pentru a permite conectarea mai multor utilizatori și a menține debitul
- Implementați de-a lungul unui număr de medii de operare majore (AIX, HP-UX, Linux, Novell NetWare, z/OS, IBM i, Solaris, Microsoft Windows)
- Vă folosiți serverul web existent, inclusiv cele de la Apache, IBM, Netscape și Microsoft.

WebSphere nu este un produs, ci o familie de trei produse care se adresează la trei piețe destinație diferite. Inima soluției WebSphere este serverul de aplicații WebSphere.

Serverul de aplicații WebSphere furnizează mediul pentru trei tipuri de obiecte. Unul este paginile server Java, care sunt analoge paginilor server active. A doua componentă conține servleturi Java și a treia este Enterprise JavaBeans. Enterprise JavaBeans este standardul care se impune pentru implementarea aplicațiilor de întreprindere pe scară foarte largă, robuste.

Aplicațiile WebSphere pot fi implementate pe aceeași platformă ca serverul web și DB2. În cazul DB2 for z/OS, DB2 Server for VM and VSE, DB2 for IBM i, WebSphere este implementat pe aceeași platformă ca produsul server DB2 Connect.

Sunt mai multe soluții WebSphere, precum și Rational Application Developer (RAD). Pentru detalii suplimentare, mergeți la <http://www.ibm.com/software/webservers/appserv/was/>

DB2 Connect ca server de aplicații Java

Multe din dezavantajele asociate cu limbaje de scriptare pot fi depășite folosind Java în schimb. IBM furnizează appleturi și aplicații care vă permit să folosiți Java la fiecare etapă a unei tranzacții web. Soluțiile IBM furnizează permisiunea pentru o combinație de tehnici, ceea ce înseamnă că puteți folosi soluții de scriptare cum ar fi Perl DBI sau Microsoft Active Server Pages cu DB2 sau mutarea spre o implementare mai robustă furnizată de un server de aplicații Java cum ar fi IBM WebSphere.

Sunt două API-uri pentru programatorii Java. Prima, JDBC, este suportată pentru folosirea Java pentru a dezvolta appleturi Java data-aware, aplicații Java precum și servleturi Java, JSP-uri (Java server pages) și EJB-uri (Enterprise Java Beans). JDBC este un API de nivel-apel (de invocare a metodei). Celălalt API Java este SQLJ. SQLJ furnizează abilitatea de a specifica SQL inline într-un program Java. DB2 poate folosi ambele API-uri, pe partea de client sau server a unei tranzacții web.

Pe partea de client sunt suportate appleturi, appleturi pregătite pentru date (data-aware) și aplicații. Pe partea bazei de date activarea Java conține obiecte bază de date, cum ar fi funcțiile definite de utilizator și proceduri stocate.

Pentru DB2 for z/OS, DB2 Server for VM and VSE și DB2 for IBM i, sunt două căi de a implementa o aplicație Java. Puteți folosi conectivitatea directă furnizată de DB2 Connect Personal Edition cu TCP/IP sau puteți alege să treceți printr-un produs server DB2 Connect care va furniza conectivitate la serverul de date mainframe IBM.

În ambele cazuri, utilizatorul pe web nu necesită software special pentru a accesa baza de date, doar un browser standard de Web. Singur lucru care trebuie instalat într-un produs server DB2 Connect și orice server web standard. Dacă serverul de Web și DB2 Connect nu se află pe aceeași mașină fizică, trebuie să fie instalat un IBM Data Server pe serverul de Web.

Pentru DB2 for z/OS, componenta cheie este un produs server DB2 Connect care rulează pe un server intermediar (mid-tier). Această componentă activează serverul JDBC, în afară de

conectarea la serverul DB2 for z/OS, DB2 Server for VM and VSE și DB2 for i. Din nou, nu este nevoie de un software special pentru browser-ul web al clientului.

IBM furnizează suport extins și unelte pentru dezvoltarea aplicațiilor și appleturilor Java. Pentru dezvoltarea aplicațiilor bază de date, DB2 Database Enterprise Developer Edition furnizează Rational Web Developer, IBM Data Studio, DB2 WebSphere Application Server, precum și produsul DB2 și DB2 Connect pentru testare. Uneltele terță parte, cum ar fi NetBeans, Borland JBuilder sau Symantec Visual Cafe, vor funcționa de asemenea cu soluțiile de bază de date ale IBM.

DB2 Connect pe serverul de Web

IBM furnizează servere HTTP (Web) cu toate produsele DB2 Connect. Produsele serverDB2 Connect, cum ar fi DB2 Connect Enterprise Edition, furnizează suport out-of-the-box pentru servere web Apache sau Lotus Domino Go și pot lucra de asemenea cu orice alt server web cum ar fi Microsoft Internet Information Server sau Netscape Enterprise Server.

Dacă lucrați cu familia de baze de date DB2 care rulează pe sisteme zSeries, iSeries, VM și VSE, un produs server DB2 Connect este necesar pe serverul de Web. Produsele server DB2 Connect vor furniza bibliotecile și interfețele de comunicație pentru a permite serverelor web să acceseze aceste platforme mainframe IBM. TCP/IP poate fi folosit pentru a comunica între serverul web și o bază de date care rulează pe System z, IBM Power Systems, VM sau VSE.

Notă: Soluțiile de web IBM furnizează posibilitatea să se lucreze cu baze de date multiple în același script Common Gateway Interface (CGI) (cum ar fi PHP) sau în aceeași tranzacție într-un script CGI.

Procedurile memorate

Un considerent important pentru aplicațiile web, ca și în lumea client/server, este să minimizezi traficul care se produce între serverul HTTP și baza de date de final. Acest considerent este important în special în procesarea tranzacțională de mare volum, care este inima celor mai multe aplicații e-business.

Abordarea recomandată este combinarea programării aplicațiilor CGI cu programarea și logica operațională încapsulată în procedurile memorate. DB2 Database for Linux, UNIX, and Windows și DB2 for z/OS, DB2 for IBM i și DB2 pentru VSE toate partajează aceeași convenție de parametru pentru invocarea procedurilor stocate.

Ca și pentru scripturile de interfață web obișnuite, browser-ul de Web trimite formularul la serverul de Web, unde este rulat scriptul de interfață web. Totuși, în locul transmiterii fiecărei instrucțiuni individuale SQL la baza de date DB2, este trimisă o cerere de executare o procedură memorată. Această procedură memorată încapsulează un număr de instrucțiuni SQL care altfel ar fi rulate individual. Procedurile memorate reduc numărul de mesaje care se duc înainte și înapoi între scriptul de interfață web și baza de date.

Avantajul cheie al procedurilor memorate este traficul de rețea scăzut între serverul HTTP și baza de date back-end DB2.

DB2 Connect și serverele de aplicații

Apariția aplicațiilor client-server a permis proiectanților de aplicații să îmbunătățească gradul de folosire și să scadă costurile de instruire furnizând aplicații cu interfețe grafice de utilizator pe platforme cum ar fi Windows. În același timp, a permis flexibilitatea delegării funcției de gestionare a bazei de date serverelor robuste de bază de date de pe o varietate de sisteme de operare și platforme hardware.

Modelul client-server, în care aplicația logică este distribuită la stațiile de lucru client este numit de obicei *client-server pe 2 niveluri*. În modelul pe 2 niveluri, aplicația este implementată pe nivelul client, iar serverul bazei de date implementează serverul (nivelul back-end). DB2 Connect furnizează suport complet pentru aplicații client-server pe 2 niveluri (2-tier), unde serverele bază de date sunt DB2 for z/OS, DB2 for IBM i sau DB2 Server for VM and VSE.

O dată cu creșterea în mărime a aplicațiilor client-server, devine evident că modelul pe 2 niveluri client-server are limitări semnificative. Distribuirea cantităților mari de informații logice la sute sau chiar mii de stații de lucru client a făcut ca gestionarea schimbărilor să fie o sarcină complexă și costisitoare. Orice schimbare în regulile de activitate necesită înlocuirea părții client a aplicației. Adesea, trebuia ca aceste modificări ale aplicației să fie făcute pe toate stațiile de lucru client din întreprindere în același timp, pentru a se asigura că regulile de activitate sunt aplicate identic.

Alt dezavantaj al modelului client-server pe 2 niveluri care devine evident o dată cu creșterea dimensiunii este cantitatea resurselor consumate de asemenea aplicații. Implementarea de sute sau mii de *clienți grași*, așa cum sunt numiți adesea clienții pe 2 niveluri, a dus la creșterea cerințelor de putere de procesare și de capacitate pentru fiecare stație de lucru client. În plus, au crescut mult și cerințele privind serverul de bază de date, deoarece fiecare client necesită o conexiune dedicată la baza de date și resursele pe care le implică menținerea unei astfel de conexiuni. În timp ce dependența modelului client-server pe 2 niveluri de distribuirea logicii de activitate poate fi oarecum redusă prin folosirea extensivă a procedurilor memorate, celelalte dezavantaje nu pot fi rezolvate cu ușurință fără a modifica modelul.

O soluție de server de aplicații

Cum costul și complexitatea aplicațiilor client-server pe 2 niveluri a crescut, majoritatea aplicațiilor mari au trecut la modelul client-server multinivel. În modelul multinivel, rolul nivelului bază de date rămâne neschimbat. Nivelul client este însă suplimentat cu unul sau două niveluri intermediare; de obicei se folosește unul singur astfel că modelul este numit *pe 3 niveluri*.

În modelul pe 3 niveluri, clientul se ocupă de tratarea interacțiunilor utilizatorului și nu conține vreo logică de activitate. Nivelul de mijloc constă dintr-unul sau mai multe servere de aplicații. Obiectivul serverului de aplicații este acela de a furniza o implementare robustă, eficientă din punct de vedere al costului a logicii din spatele proceselor și regulilor de activitate. Ca și în cazul modelului pe 2 niveluri, implementarea regulilor de activitate este adesea suplimentată prin folosirea procedurilor memorate pentru a îmbunătăți performanța.

Deoarece stațiile de lucru client nu mai implementează logica aplicației și se ocupă doar de tratarea interacțiunilor de utilizator, cererile de resurse pentru nivelul client sunt cu mult reduse. De fapt, nivelul client din modelul pe 3 niveluri este adesea numit *client subțire*. În plus, deoarece un server de aplicații centralizat tratează cererile de la toți clienții, are capacitatea de a partaja resurse, cum ar fi conexiunile la baza de date, între toți clienții. Ca rezultat, serverul de bază de date nu mai trebuie să mențină conexiuni dedicate pentru fiecare utilizator de aplicație.

Se pot da mai multe exemple de folosire a serverelor de aplicații pe 3 niveluri în prezent. Aproape toți furnizorii Enterprise Resource Planning (ERP) își implementează aplicațiile folosind modelul pe 3 niveluri, cum ar fi aplicațiile SAP R/3 și PeopleSoft V7. Alte exemple sunt principalii furnizori Enterprise Relationship Management, cum ar fi Siebel și Vantive.

Servere de aplicații și DB2 Connect

Produsele server DB2 Connect furnizează suport de înțelegere pentru implementarea aplicațiilor multinivel. Suportul furnizat de DB2 Connect include o varietate de API-uri care pot fi folosite pentru a dezvolta logică aplicație (ODBC, ADO.NET,

DB2 CLI, SQL încorporat, JDBC, SQLJ, Perl, PHP și OLE DB), precum și o infrastructură completă de comunicații pentru interacționarea cu servere de bază de date din familia DB2.

DB2 Connect suportă de asemenea implementări în care un nivel (tier) de bază de date conține mai multe servere de bază de date din familia DB2. Aceasta permite serverelor de aplicații să implementeze tranzacții ce actualizează date ce se află pe mai multe servere de bază de date într-o singură tranzacție.

Suportul protocol comitere în două faze furnizat de DB2 Connect se asigură de integritatea unor asemenea tranzacții distribuite. De exemplu, o aplicație poate actualiza date dintr-o bază de date DB2 for z/OS și DB2 Database for Linux, UNIX, and Windows în aceeași tranzacție. Dacă suportul cerere distribuită este instalat și activat, aplicația poate citi o actualizare bază de date Oracle o bază de date din familia DB2 în aceeași tranzacție.

În următoarea diagramă, API-urile precum și mecanismul de conectivitate dintre serverul de aplicații și serverele de bază de date back-end este furnizat de un produs server DB2 Connect, cum ar fi DB2 Connect Enterprise Edition.

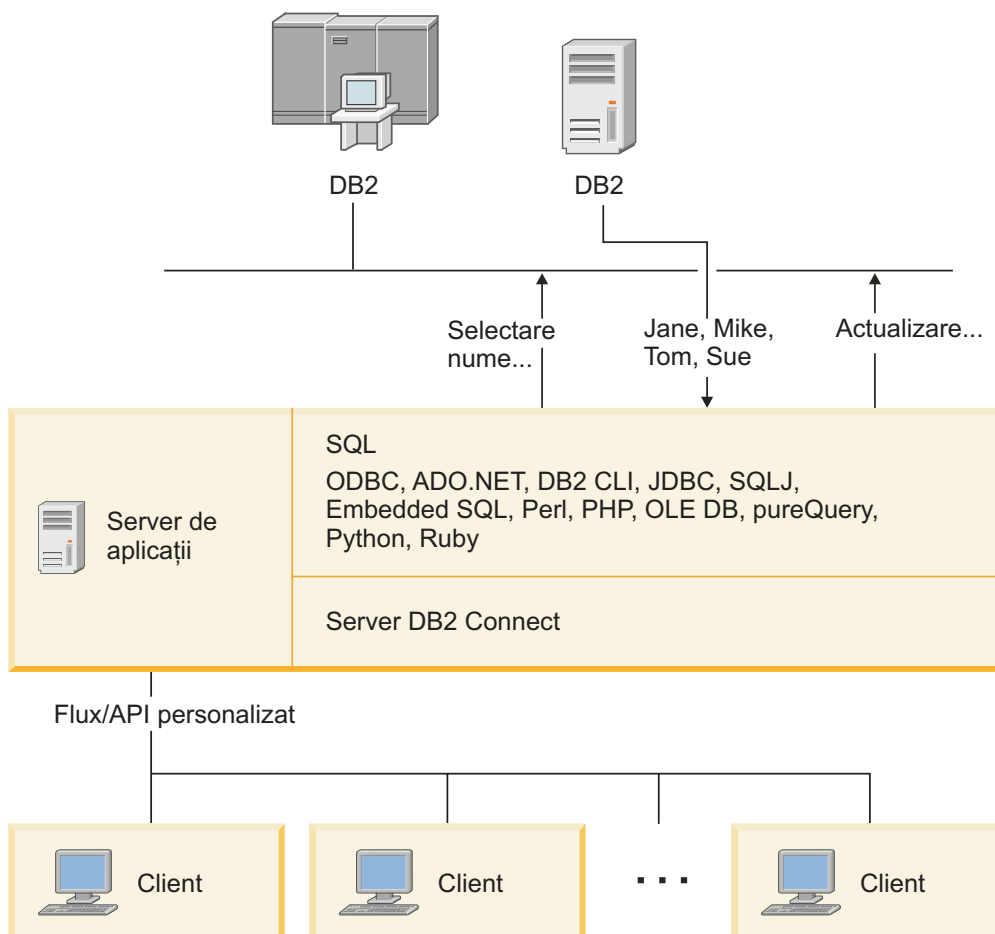


Figura 6. Suport DB2 Connect pentru servere de aplicații

Caracteristicile avansate ale DB2 Connect, cum ar fi punerea în pool a conexiunii, reduc cu mult cererile de resurse ale aplicației și simplifică implementarea serverului de aplicații.

DB2 Connect și configurațiile serverului de aplicații

Un produs server DB2 Connect este necesar pentru folosirea cu servere de aplicații. DB2 Connect Personal Edition nu este suportat și nu este licențiat pentru folosirea cu servere de aplicații. În plus, clienții care implementează servere de aplicații ar trebui să revadă termenii și condițiile furnizate cu copia lor de DB2 Connect pentru a înțelege numărul de licențe utilizator care trebuie obținute.

Sunt două metode de implementare pentru DB2 Connect în mediul server de aplicații. Un produs server DB2 Connect poate fi instalat pe oricare din:

- Mașina server de aplicații
- O mașină server de comunicații separată

În majoritatea situațiilor, instalarea unei copii de DB2 Connect pe același server cu serverul de aplicații este soluția preferată. Instalarea DB2 Connect pe serverul aplicație vă permite să participați la orice schemă preluare eroare și echilibrarea încărcării pe care o poate implementa un server de aplicații. Această setare poate oferi un nivel mai bun de performanță, deoarece elimină un hop de rețea suplimentar, necesar când se instalează DB2 Connect pe un server separat. Mai mult, administrarea poate fi simplificată, deoarece nu este nevoie de instalarea și întreținerea unui server suplimentar.

Instalarea DB2 Connect pe un server separat este o opțiune bună în situațiile unde produsul dumneavoastră server DB2 Connect nu este disponibil pentru sistemul de operare sau platforma hardware unde rulează serverul de aplicații.

DB2 Connect și monitoarele de procesare a tranzacțiilor

Un server de aplicații permite unui număr mare de utilizatori să execute aplicații folosind un minim de resurse de sistem. Un server de aplicații poate fi extins pentru a permite invocarea tranzacțiilor coordonate din aplicații executate de serverul de aplicații. Această coordonare a tranzacțiilor este cunoscută sub numele de monitor TP (Transaction Processing - Procesare tranzacții). Un monitor TP lucrează împreună cu un server de aplicații.

Ne putem gândi la o *tranzacție* ca la un eveniment de rutină, de obicei o cerere de serviciu, în rularea operațiilor zilnice ale unei organizații. Monitoarele TP au fost proiectate pentru a permite procesarea în ordine a tranzacțiilor.

Procesarea tranzacțiilor

Fiecare organizație are reguli și proceduri ce descriu modul în care ar trebui să se lucreze. Aplicațiile de utilizator care implementează aceste reguli pot fi numite *logica operațională*. Tranzacțiile pe care aceste aplicații operaționale le execută sunt adesea numite procesare de tranzacții (Transaction Processing) sau procesare de tranzacții online (Online Transaction Processing - OLTP).

Caracteristicile cheie ale OLTP într-un mediu comercial sunt:

Mulți utilizatori

De obicei procesarea tranzacțiilor este folosită de majoritatea persoanelor dintr-o organizație, deoarece starea curentă a activității este afectată de multe persoane.

Repetitivitate

Majoritatea interacțiunilor cu calculatorul tind să se reducă la un proces care se tot repetă. De exemplu, introducerea unei comenzi și procesarea plăților sunt realizate de foarte multe ori în fiecare zi.

Interacțiuni scurte

Majoritatea interacțiunilor pe care persoanele dintr-o organizație le au cu sistemul de procesare a tranzacțiilor durează puțin.

Date partajate

Deoarece datele reprezintă starea organizației, poate exista un singur set de date.

Integritatea datelor

Datele trebuie să reprezinte starea curentă a organizației și trebuie să fie consistente intern. De exemplu, fiecare comandă trebuie să fie asociată cu o înregistrare de cumpărător.

Cost scăzut/tranzacție

Deoarece procesarea tranzacțiilor reprezintă un cost direct al activității comerciale, costul sistemului trebuie să fie minim. DB2 Connect permite aplicațiilor de sub controlul serverului de aplicații ce rulează pe Linux, UNIX și Windows să execute tranzacții pe LAN la distanță, pe serverele de bază de date mainframe IBM, aceste tranzacții fiind coordonate de un monitor TP.

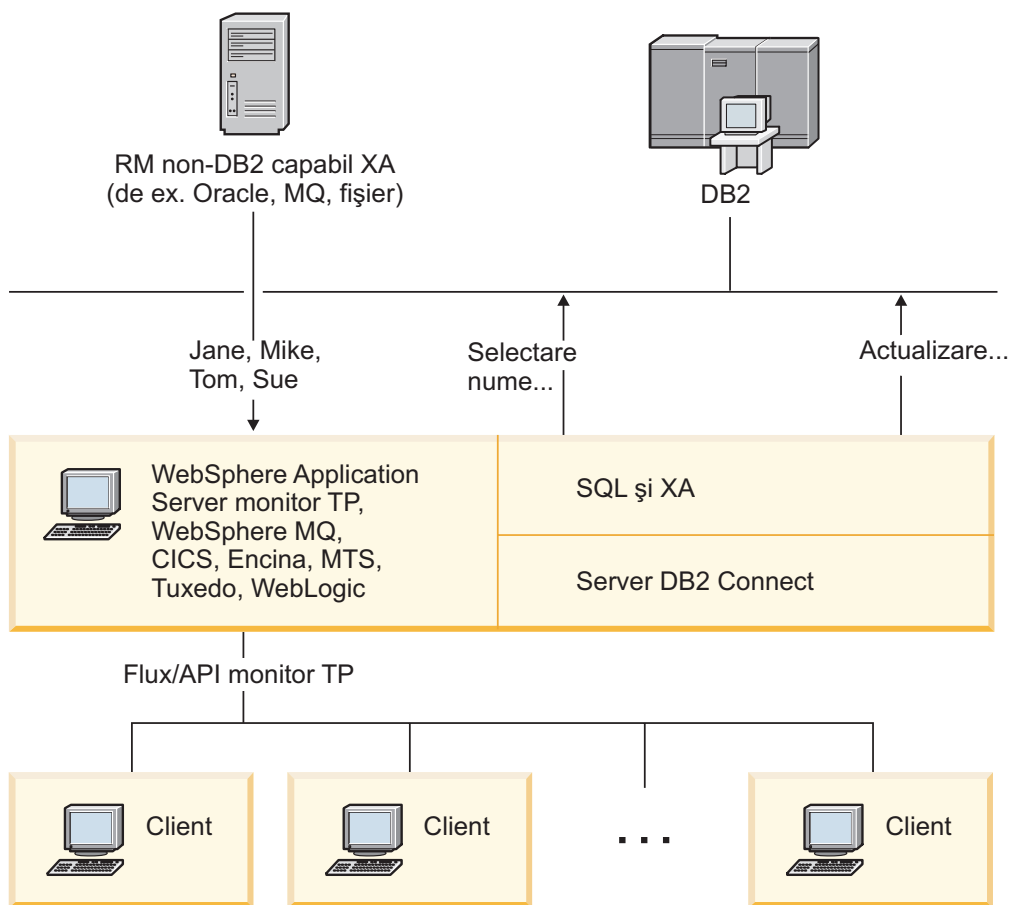


Figura 7. Suportul DB2 Connect pentru monitoare TP

În Figura 7, API-urile, precum și mecanismul de conectivitate dintre serverul de aplicații și serverele de bază de date back-end sunt furnizate de un produs server DB2 Connect, cum ar fi DB2 Connect Enterprise Edition.

Exemple de monitoare pentru procesarea tranzacțiilor

În prezent, cele mai răspândite monitoare TP de pe piață sunt:

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

În tranzacții coordonate de aceste monitoare TP pot fi folosite servere de bază de date la distanță IBM Power Systems, System z și din rețeaua locală.

Modelul X/Open DTP (Distributed Transaction Processing)

Pentru a actualiza mai multe resurse într-o singură tranzacție, ar putea fi necesară o aplicație care să execute logica operațională. De exemplu, o aplicație bancară care implementează un transfer de bani de la un cont la altul poate cere debitarea unei baze de date (contul "sursă") și depozitarea în altă bază de date (contul "destinație").

Este de asemenea posibil ca aceste două baze de date să provină de la furnizori diferiți. De exemplu, o bază de date poate fi DB2 for z/OS, iar cealaltă poate fi o bază de date Oracle. În loc să se implementeze în fiecare monitor TP fiecare interfață de tranzacții de proprietar, a fost definită o interfață comună de tranzacții între un monitor TP și orice resursă accesată de o aplicație. Această interfață este numită *interfața XA*. Un monitor TP care folosește interfața XA este un *manager de tranzacții (TM - Transaction Manager) compatibil XA*. O resursă actualizabilă care implementează interfața XA este un *manager de resurse (RM - Resource Manager) compatibil XA*.

Toate monitoarele TP menționate mai sus sunt manageri de tranzacții compatibili XA. Când sunt accesate via DB2 Connect, bazele de date gazdă la distanță, IBM Power Systems și bazele de date DB2 bazate pe LAN sunt manageri de resurse (RM) compatibili XA. De aceea, orice monitor TP care are un manager de tranzacții compatibil XA poate folosi baze de date gazdă, IBM Power Systems și DB2 bazate pe LAN în aplicațiile de întreprindere care execută tranzacții.

Partea 2. Referințe DB2 Connect

Capitolul 4. Actualizarea directorilor bazei de date

DB2 Connect folosește următoarele directoare pentru a gestiona informațiile de conectare la baza de date:

- *director de bază de date de sistem*, care conține numele, nodul și informațiile de autentificare pentru fiecare bază de date pe care o accesează DB2 Connect.
- *director de noduri*, care conține adresa de rețea și informațiile protocolului de comunicație pentru fiecare server de bază de date mainframe IBM pe care îl accesează DB2 Connect.
- *director servicii conexiune bază de date (DCS)*, care conține informații specifice bazelor de date ale serverelor de bază de date mainframe IBM.

Notă:

1. Înainte de a actualiza aceste directoare, ar trebui să configurați comunicațiile de pe serverele bază de date mainframe IBM și stațiile de lucru.
2. Directoarele bazei de date pot fi actualizate folosind Configuration Assistant (CA).

Pentru a actualiza directoarele bazei de date:

1. Colectați informațiile directorului bazei de date folosind fișa de lucru pentru personalizarea directorului
2. Vedeți subiectul “Actualizarea directorilor cu informații despre mașinile server de bază de date” din Control Center

Valorile din directorul bazei de date de sistem

Pentru fiecare instanță a managerului de bază de date există un director de bază de date de sistem, care conține câte o intrare pentru fiecare bază de date care a fost catalogată pentru instanța respectivă. În produsele DB2 Connect, directorul de bază de date de sistem conține informații privind numele, aliasul, numele de nod și tipul de autentificare ale fiecărei baze de date.

În directorul bazei de date de sistem puteți specifica următoarele informații:

Nume bază de date

Valoarea pe care ați scris-o în tabela Parametri director DCS.

Alias bază de date

Un alias pentru serverul bază de date mainframe IBM. Acest nume va fi utilizat de orice program de aplicație care accesează baza de date. Implicit, este utilizată valoarea pe care o specificați pentru numele bazei de date.

Format: 1–8 caractere alfanumerice pe un singur octet, inclusiv semnul pentru numere (#), semnul a rond (@), semnul dolar (\$) și liniuță de subliniere (_). Nu poate începe cu o liniuță de subliniere sau cu un număr.

Nume nod

Valoarea pe care ați scris-o în tabela Parametri director nod.

Autentificare

Specifică unde va fi făcută validarea numelui de utilizator și a parolei pentru conexiunile inițiate de pe serverul DB2 Connect. Opțiunile valide sunt: SERVER, SERVER_ENCRYPT, CLIENT, KERBEROS, SERVER_ENCRYPT_AES și DATA_ENCRYPT. Nu există suport pentru tipul de autentificare GSSPLUGIN în directorul de bază de date de sistem.

Valorile directorului de noduri

În directorul de noduri puteți specifica următoarele informații:

Nume nod

O poreclă pentru sistemul server bază de date mainframe IBM pe care se află baza de date la distanță. Acest nume este definit de utilizator. Scrieți același nume de nod atât în tabela de parametri director nod, cât și în tabela de parametri director de baze de date sistem.

Format: 1–8 caractere alfanumerice pe un singur octet, inclusiv semnul pentru numere (#), semnul a rond (@), semnul dolar (\$) și liniuță de subliniere (_). Nu poate începe cu o liniuță de subliniere sau cu un număr.

Protocolul

Trebuie să fie TCP/IP.

Tipul de securitate

Tipul verificărilor de securitate care vor fi efectuate. Pentru nodurile TCP/IP, SECURITY SOCKS este o opțiune care specifică faptul că nodul va fi activat pentru SOCKS, caz în care variabilele de mediu SOCKS_NS și SOCKS_SERVER sunt obligatorii și trebuie setate astfel încât să activeze SOCKS.

Numele TCP/IP al gazdei la distanță sau adresa IP

La definirea nodurilor TCP/IP, fie numele gazdei TCP/IP la distanță, fie adresa TCP/IP la distanță. Dacă este specificat un nume de gazdă, atunci acesta trebuie rezolvat la stația de lucru DB2 Connect, fie prin căutarea DNS (Domain Name Server), fie printr-o intrare în fișierul de gazde TCP/IP locale.

Pentru gazdele la distanță DB2 for z/OS, numele gazdei apare în mesajul DSNL004I (DOMAIN=hostname) când este pornit Distributed Data Facility (DDF). Poate fi folosită și comanda -DISplay DDF.

La accesarea unui grup de partajare z/OS, numele domeniului ar trebui să se mapeze pe adresa VIPA dinamică a grupului DB2. Această adresă rutează cel mai puțin încărcat membru DB2. Pentru a accesa un anumit număr folosiți adresa specifică VIPA membru dinamic DB2 și opriți rutarea Sysplex. Fiecare mesaj DSNL004I de membru afișează numele de domeniu specific al membrului.

Numele serviciului TCP/IP sau numărul portului

La definirea nodului TCP/IP, fie numele serviciului TCP/IP la distanță, fie numărul portului. Acestea trebuie definite pentru TCP/IP la gazda de la distanță. Numărul de port 446 a fost înregistrat ca numărul implicit de port pentru DRDA.

Pentru gazde la distanță DB2 for z/OS, numărul portului este definit în BSDS ca PORT și este de asemenea furnizat în mesajul DSNL004I (TCPPOrt=portnumber) când este pornit DDF. Poate fi folosită și comanda -DISplay DDF.

La accesarea unui grup de partajare z/OS, numele domeniului ar trebui să se mapeze pe adresa VIPA dinamică a grupului DB2. Această adresă rutează cel mai puțin încărcat membru DB2. Pentru a accesa un anumit număr folosiți adresa specifică VIPA membru dinamic DB2 și opriți rutarea Sysplex. Fiecare mesaj DSNL004I de membru afișează numele de domeniu specific al membrului.

Notă: Un al doilea port folosit pentru operații de resincronizare comitere în două faze pe conexiuni TCP/IP poate fi asignat de server. De exemplu, setul de date bootstrap DB2 for z/OS alocă un număr de port (RESPORT) pentru a fi folosit pentru resincronizare pentru conexiuni de ieșire doar pentru DB2 for z/OS. Pentru aceasta nu este necesară definirea unui nume de serviciu.

Valorile din directorul DCS

Puteți specifica următoarele informații în directorul DCS:

Nume bază de date

O poreclă definită de utilizator pentru serverul bază de date mainframe IBM. Scrieți același nume de bază de date atât în tabela de parametri director DCS, cât și în tabela de parametri director baze de date sistem.

Format: 1–8 caractere alfanumerice pe un singur octet, inclusiv semnul pentru numere (#), semnul a rond (@), semnul dolar (\$) și liniuță de subliniere (_). Nu poate începe cu o liniuță de subliniere sau cu un număr.

Nume bază de date destinație

Baza de date de pe sistemul server de bază de date mainframe IBM, după cum urmează:

System z

Un subsistem DB2 for z/OS identificat de LOCATION NAME sau unul dintre numele LOCATION alias definite pe serverul z/OS.

LOCATION NAME (nume locație) poate fi determinat prin logarea pe TSO și lansarea următoarei interogări SQL utilizând una dintre uneltele de interogare disponibile:

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

De asemenea, sunt definite mai multe LOCATION NAME în Boot Strap Data Set (BSDS), precum și în mesajul DSNL004I (LOCATION=locație), care este scris când pornește Distributed Data Facility (DDF). Poate fi folosită și comanda -DISplay DDF.

La accesarea unui grup de partajare z/OS, numele domeniului ar trebui să se mapeze pe adresa VIPA dinamică a grupului DB2. Această adresă rutează cel mai puțin încărcat membru DB2. Pentru a accesa un anumit număr folosiți adresa specifică VIPA membru dinamic DB2 și opriți rutarea Sysplex. Fiecare mesaj DSNL004I de membru afișează numele de domeniu specific al membrului.

VSE sau VM

Numele bazei de date (DBNAME)

IBM Power Systems

Numele bazei de date relaționale (RDBNAME)

Altul Pentru sisteme de operare Windows, Linux și UNIX, aliasul bază de date găsit în directorul bază de date.

Șir de parametri

Dacă doriți să schimbați valorile implicite, specificați unii sau toți parametrii următori, în această ordine.

fișier-mapare

Numele unui fișier de mapare SQLCODE care înlocuiește maparea implicită SQLCODE. Pentru a opri maparea SQLCODE, specificați **NOMAP**.

Notă: La procesarea unei cereri de interogare, serverul DRDA returnează date sub forma unui set de rânduri care reprezintă setul rezultat. Cu fiecare rând, se întoarce de asemenea un SQLCA, în mod normal conținând un cod SQL zero sau pozitiv (cum ar fi +12 sau +802). Dacă folosiți un fișier de mapare personalizat la un

server DB2 Connect, asemenea coduri SQL pozitive nu vor fi mapate dacă sunt conținute într-un fișier de mapare personalizat și au mapări personalizate (de exemplu, sunt mapate la un cod sql diferit sau au jetoane de mapare personalizate).

Este important de subliniat că:

1. Codurile SQL pozitive reprezintă avertismente, opuse codurilor SQL negative care indică condiții de eroare. Toate codurile SQL negative vor fi mapate întotdeauna în toate circumstanțele, indiferent de fișierul de mapare care este utilizat. Toate codurile SQL pozitive, conținute în fișierul de mapare personalizat și mapate în ele însele fără nici o modificare, vor fi mapate întotdeauna în acest mod. De asemenea, vor fi de asemenea mapate codurile SQL pozitive care nu sunt conținute în fișierul de mapare personalizat de pe serverul DB2 Connect.
2. Dacă utilizați fișierul de mapare implicit sau dacă vă conectați direct la baza de date gazdă, maparea codurilor SQL va fi efectuată întotdeauna pentru toate codurile SQL.

,D Acesta este parametrul din poziția a doua. Dacă este specificat că aplicația se va deconecta de la baza de date a serverului de bază de date mainframe IBM când este returnat unul din următoarele SQLCODE-uri:

SQL3000N
SQL30040N
SQL30050N
SQL30051N
SQL30053N
SQL30060N
SQL30070N
SQL30071N
SQL30072N
SQL30073N
SQL30074N
SQL30090N

Atunci când nu este specificat parametrul de deconectare **,D**, o deconectare va fi efectuată doar când se returnează următoarele coduri SQL (SQLCODE):

SQL30020N
SQL30021N
SQL30041N
SQL30061N
SQL30081N

Pentru explicarea acestor coduri, consultați *Referințe mesaje*.

Notă: Dacă DB2 Connect se deconectează din cauza unei erori, va fi efectuată în mod automat o refacere.

„INTERRUPT_ENABLED

Acesta este parametrul din poziția a treia.

INTERRUPT_ENABLED (întrerupere activată) se aplică doar dacă serverul terminal nu suportă întreruperi. Dacă un server suportă întreruperea DRDA, fluxul DB2 Connect va transmite pur și simplu cererea de întrerupere serverului.

Dacă INTERRUPT_ENABLED este configurat în directorul DCS de pe stația de lucru DB2 Connect și o aplicație client lansează o întrerupere în timp ce este conectată la un server de bază de date mainframe IBM, DB2 Connect va realiza întreruperea abandonând conexiunea și derulând înapoi unitatea de lucru. Acest comportament de întrerupere este suportat pe AIX și Windows.

Aplicația va recepționa codul SQL (-30081), care indică terminarea conexiunii cu serverul. Aplicația trebuie apoi să stabilească o nouă conexiune cu serverul bază de date mainframe IBM, pentru a procesa cereri suplimentare ale bazei de date. Pe alte platforme decât AIX V5.2 și mai recente și Windows, DB2 Connect nu suportă opțiunea de deconectare automată când o aplicație care îl folosește primește o cerere de întrerupere.

Notă: Acest suport funcționează pentru conexiunile TCP/IP pe orice platforme. Clientul ar putea închide socket-ul, dar - în funcție de implementarea serverului- dar se poate sau nu să existe o recepționare remarcabilă. DB2 for z/OS utilizează apeluri socket asincrone și de aceea este capabil să detecteze pierderea conexiunii și refacerea oricăror instrucțiuni SQL cu rulare îndelungată care sunt în desfășurare.

,,,,,SYSPLEX

Acest parametru, din poziția a șasea, poate fi folosit pentru a activa explicit suport DB2 Connect SYSPLEX pentru o anumită bază de date.

,,,,,LOCALDATE="*<valoare>*"

Acest parametru, din poziția a șaptea, este folosit pentru a activa suport pentru formatare date DB2 Connect. Este implementat folosind o mască de dată pentru *<valoare>*, după cum urmează:

Presupunând că ați lansat următoarele instrucțiuni CLP (procesor linie de comandă):

```
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
```

Aliasul bazei de date *newyork1* va fi utilizat pentru accesarea unei baze de date gazdă fără transformarea datelor, deoarece nu a fost specificată nici o mască de dată.

Însă cu noul suport de formatare a datei, puteți utiliza acum următoarele comenzi CPL. În acest caz, deoarece se folosește CLP și șirul de parametri este specificat folosindu-se marcajul cu ghilimele, valoarea LOCALDATE trebuie să fie specificată între două perechi de ghilimele. Observați folosirea caracterului escape "\" (backslash) al sistemului de operare, pentru a asigura păstrarea ghilimelelor în specificația LOCALDATE.

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

Aliasul bazei de date *newyork2* vă furnizează accesul la aceeași bază de date gazdă, dar în plus, are specificată o mască de format al datei. Acest exemplu ilustrează faptul că masca de format al

datei este specificată prin utilizarea cuvântului cheie LOCALDATE, fiind parametrul din poziția a șaptea în câmpul PARMS al unei intrări de director DCS.

Pentru ca masca de dată să fie corectă, TOATE dintre următoarele trebuie să fie adevărate:

1. Poate fi doar cel mult cel mult o secvență de fiecare tip Y-uri, M-uri și D-uri, unde Y este o cifră pentru an, M este o cifră pentru lună și D este o cifră pentru zi.
2. Numărul maxim de Y într-o secvență este 4.
3. Numărul maxim de M într-o secvență este 2.
4. Numărul maxim de D într-o secvență este 2.

De exemplu, toate măștile de dată următoare sunt corecte:

"YYYYMmDd" - cifrele Y, M și D nu sunt sensibile la majuscule
"MM+DD+YYYY" - este corect să fie o mască mai lungă de 10 octeți și să aibă caractere altele decât Y, M și D în mască
"abcYY+MM" - este corect să nu existe o secvență de D-uri

Următoarele măști de date nu sunt valide:

"YYYYyMMDD" - invalidă deoarece sunt 5 de Y într-o secvență
"YYYYMDDM" - incorectă deoarece sunt două secvențe de M-uri

Dacă o mască de format de dată este incorectă, nu se va produce nici o eroare. Pur și simplu va fi ignorată. Dacă o mască de dată este corectă, aceasta nu înseamnă că va fi neapărat utilizată. Transformarea formatului de dată bazat pe o mască de date corectă va fi efectuat doar dacă TOATE dintre următoarele sunt adevărate:

1. Nu este nici o eroare SQL.
2. Ieșirea este o valoare de dată în format compatibil ISO (ISO și JIS).
3. Zona de date de ieșire are o lungime de cel puțin 10 octeți. Aceasta este lungimea minimă a unei zone de date de ieșire pentru ca o valoare de date să fie stocată acolo, chiar dacă NU este efectuată nici o transformare de format de dată calendaristică. Această cerință se aplică chiar dacă masca de format pentru data calendaristică se termină după mai puțin de 10 octeți.
4. Există o mască validă de format de dată calendaristică, specificată în intrarea de director DCS și această mască încapă în zona de date de ieșire.

,,,,,,BIDI=<ccsid>

Acest parametru, din poziția a noua, este utilizat pentru specificarea CCSID-ului bidirecțional (BiDi) care să fie utilizat pentru a înlocui CCSID-ul BiDi implicit al serverului de baze de date. De exemplu:

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

unde xyz reprezintă CCSID-ul înlocuit.

Fișă de lucru pentru personalizarea directorului

Fișa de lucru pentru personalizarea directorului vă arată ce informații trebuie să colectați. Ați putea să considerați convenabil să faceți o copie a fișei de lucru și să introduceți valorile sistemului dumneavoastră.

Parametrii directorului de noduri

Tabela 1. Parametrii directorului de noduri

Parametru	Exemplu	Valoarea dumneavoastră
Numele nodului	DB2NODE	
Nume gazdă la distanță (nod TCP/IP)	ZOSHOST	
Server (nume serviciu și număr port TCP/IP)	db2inst1c (sau 446)	

Notă:

1. Numărul de port TCP/IP implicit pentru DRDA este 446
2. Dacă nu știți că serverul bază de date mainframe IBM suportă SECURITY SOCKS, nu specificați SECURITY pentru un nod TCP/IP.

Parametrii directorului DCS

Tabela 2. Parametrii directorului DCS

Parametru	Exemplu	Valoarea dumneavoastră
Nume bază de date	DB2DB	
Nume bază de date destinație	NEW_YORK3	
Solicitant aplicație		
Șir de parametri	" ,,,,,,LOCALDATE=\\\"YYMMDD\\\"\\\""	

Parametrii directorului bază de date sistem

Tabela 3. Parametrii directorului bază de date sistem

Parametru	Exemplu	Valoarea dumneavoastră
Nume bază de date	DB2DB	
Alias bază de date	NYC3	
Numele nodului	DB2NODE	
Autentificare	SERVER	

Definirea mai multor intrări pentru aceeași bază de date

Pentru fiecare bază de date, trebuie să definiți cel puțin o intrare în fiecare din cele trei directoare (directorul de noduri, directorul DCS și directorul de baze de date sistem). În unele cazuri, ați putea dori să definiți mai mult de o intrare pentru baza de date.

De exemplu, ați putea vrea să opriți maparea SQLCODE pentru aplicații care au fost portate de la serverul bază de date mainframe IBM, dar să acceptați maparea implicită pentru aplicațiile care au fost dezvoltate pentru mediul client/server. Veți face aceasta după cum urmează:

- Definiți o intrare în directorul de noduri.
- Definiți două intrări în directorul DCS, cu nume diferite de baze de date. Pentru o intrare specificați NOMAP în șirul de parametri.
- Definiți două intrări în directorul de baze de date sistem, cu aliasuri diferite de baze de date și cele două nume de baze de date pe care le-ați specificat în directorul DCS.

Ambele aliasuri accesează aceeași bază de date, unul cu mapare SQLCODE, iar celălalt fără mapare SQLCODE.

Tratarea datelor BiDi

Următoarea secțiune se aplică doar la serverele z/OS. Această caracteristică nu trebuie activată pentru un server DB2 for IBM i, deoarece suportul complet BiDi este deja furnizat.

Următoarele atribute BiDi sunt necesare pentru o tratare corectă a datelor BiDi pe diferite platforme:

- Formă numerică (ARABIC versus HINDI)
- Orientare (RIGHT-TO-LEFT versus LEFT-TO-RIGHT)
- Shaping (SHAPED versus UNSHAPED)
- Schimbare simetrică (DA sau NU)
- Tip text (LOGICAL versus VISUAL)

Deoarece valorile implicite pe diferite platforme nu sunt aceleași, apar probleme când sunt trimise date DB2 de pe o platformă pe alta. De exemplu, platformele Windows folosesc date LOGICAL UNSHAPED, în timp ce datele z/OS sunt în format SHAPED VISUAL. De aceea, fără suport pentru atribute BiDi, datele trimise de la DB2 for z/OS la DB2 Connect pe Windows se afișează incorect.

Când se schimbă date între DB2 Connect și o bază de date de pe un server, receptorul este cel care, în mod normal, efectuează conversia asupra datelor de intrare. Aceeași convenție se aplică, în mod normal și în cazul transformării dispunerii BiDi, care este suplimentară conversiei codurilor de pagină. Însă deocamdată niciun produs DB2 gazdă nu suportă CCSID-uri specifice BiDi sau transformarea dispunerii BiDi. De aceea, DB2 Connect a fost îmbunătățit cu capacitatea opțională de a efectua transformări de dispunere BiDi asupra datelor pe care le va trimite bazei de date de pe server în plus față de datele primite de la baza de date de pe server.

Pentru ca DB2 Connect să realizeze transformări de dispuneri BiDi pentru datele care pleacă spre baza de date server, trebuie să fie înlocuit CCSID-ul BiDi al bazei de date server. Aceasta se realizează prin utilizarea parametrului BIDI în câmpul PARMS al intrării directorului bazei de date DCS pentru baza de date server.

Utilizarea acestei caracteristici este cel mai bine ilustrată cu un exemplu.

Să presupunem că folosiți IBM Data Server în ebraică cu CCSID-ul 62213 (tipul de șir BiDi 5) și ați dori să accesați o bază de date DB2 gazdă cu CCSID-ul 424 (tipul de șir BiDi 4). Știți însă că datele conținute de baza de date DB2 gazdă sunt bazate pe CCSID-ul 62245 (tipul de șir BiDi 10).

Sunt două probleme în această situație. Prima este că baza de date DB2 gazdă nu face diferența între tipurile de șir BiDi cu CCSID-urile 424 și 62245. A doua problemă este faptul că baza de date DB2 gazdă nu recunoaște CCSID-ul 62213 de pe IBM Data Server. Suportă doar CCSID-ul 62209 (tipul de șir BiDi 10), care este bazat pe același cod de pagină ca și CCSID-ul 62213.

Va trebui să vă asigurați pentru început că datele trimise la baza de date gazdă DB2 sunt în formatul BiDi tip de șir 6 și de asemenea să fie anunțat DB2 Connect că trebuie să realizeze transformarea dispunerii BiDi pentru datele pe care le primește de la baza de date gazdă DB2. Veți folosi următoarea catalogare pentru baza de date gazdă DB2:

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

Aceasta anunță DB2 Connect să înlocuiască CCSID-ul 424 al bazei de date gazdă DB2 cu 62245. Această înlocuire include următoarele prelucrări:

1. DB2 Connect se va conecta la baza de date DB2 gazdă folosind CCSID-ul 62209 (tipul de șir BiDi 10).
2. DB2 Connect va realiza transformarea de dispunere BiDi pentru datele pe care urmează să le trimită către baza de date DB2 gazdă, din CCSID-ul 62213 (tipul de șir BiDi 5) în CCSID-ul 62209 (tipul de șir BiDi 10).
3. DB2 Connect va realiza transformarea de dispunere BiDi pentru datele pe care le primește de la baza de date DB2 gazdă, din CCSID-ul 62245 (tipul de șir BiDi 10) în CCSID-ul 62213 (tipul de șir BiDi 5).

Notă:

1. Pentru ca parametrul BIDI să aibă efect, va trebui să fie setată la YES variabila de mediu sau valoarea de registru **DB2BIDI**. **DB2BIDI** trebuie setat pe stația de lucru DB2 Connect unde este catalogată intrarea directorului bazei de date DCS. Pentru aplicațiile care rulează pe un client la distanță al unui server DB2 Connect, variabila **DB2BIDI** trebuie să fie setată și pe client.
2. Dacă doriți ca DB2 Connect să realizeze transformarea de dispunere pentru datele pe care urmează să le trimită către baza de date DB2 gazdă chiar dacă nu trebuie să înlocuiți CCSID-ul, trebuie să mai adăugați parametrul BIDI în câmpul PARMS din directorul de bază de date DCS. În acest caz, CCSID-ul pe care ar trebui să-l furnizați ar trebui să fie CCSID-ul bazei de date gazdă DB2 implicit.
3. În unele cazuri, folosirea unui CCSID bidirecțional ar putea cauza modificarea interogării SQL, astfel încât să nu fie recunoscută de serverul DB2. Mai precis, ar trebui să evitați să utilizați CCSID-urile IMPLICIT CONTEXTUAL și IMPLICIT RIGHT-TO-LEFT atunci când poate fi utilizat un tip de șir diferit. CCSID-urile CONTEXTUAL pot produce rezultate nepredictibile dacă interogarea SQL conține șiruri între ghilimele. Evitați să folosiți șiruri în instrucțiunile SQL și folosiți în schimb variabile gazdă, de câte ori este posibil.
Dacă un anumit CCSID bidirecțional cauzează probleme care nu pot fi corectate aplicând aceste recomandări, ar trebui să setați variabila de mediu sau valoarea de registru **DB2BIDI** la NO.

Specificațiile șirului de parametri

În continuare sunt prezentate exemple de parametri DCS (fiecare linie este un set de parametri):

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

Sau puteți să acceptați valorile implicite, fără să mai specificați un șir de parametri.

Notă: Trebuie să folosiți caracterul escape "\" (backslash) al sistemului de operare când folosiți CLP din linia de comandă a sistemului de operare pe sistemele UNIX, deoarece trebuie să fie specificate două perechi de ghilimele pentru masca LOCALDATE din șirul de parametri. De exemplu:

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

Aceasta are ca rezultat următoarea intrare de director DCS:

DCS 1 entry:

Local database name	= X
Target database name	= Y
Application requestor name	=
DCS parameters	= ,,,,,,LOCALDATE="YYMMDD"
Comment	=
DCS directory release level	= 0x0100

Capitolul 5. Securitatea DB2 Connect

Autentificarea utilizatorilor este importantă când se folosește DB2 Connect, deoarece utilizatorii pot fi atât locali, cât și la distanță față de DB2 Connect și baza de date în care se află datele pe care doresc să le acceseze. Sunt prezentate conexiunile de încredere și suportul Kerberos, precum și considerente privind securitatea pentru bazele de date de pe mașinile gazdă.

Conexiunile de încredere prin DB2 Connect

Unele servere de bază de date DB2 suportă contexte de încredere. Un *context de încredere* permite administratorului bazei de date să definească condiții sub care aplicația de client va putea să creeze o conexiune de încredere. Unei *conexiuni de încredere* îi este permis să facă lucruri pe care o conexiune normală nu le poate face.

Există două tipuri de conexiuni de încredere, implicite și explicite. Atunci când creați o conexiune, dacă obțineți o conexiune de încredere explicită sau o conexiune de încredere implicită, sau o conexiune obișnuită depinde de cum cereți o conexiune de încredere și dacă conexiunea corespunde criteriului definit în contextul de încredere pe server, așa cum este rezumat în Tabela 4.

Tabela 4. Ce tip de conexiuni rezultă din diferite combinații de acțiuni

	Conexiunea îndeplinește criteriul de server pentru a fi de încredere	Conexiunea nu îndeplinește criteriul de server pentru a fi de încredere
Ați cerut să fie de încredere conexiunea	Conexiune de încredere explicită	Conexiune obișnuită și este întors un avertisment SQL20360W (SQLSTATE 01679).
Nu ați cerut să fie de încredere conexiunea	Conexiune de încredere implicită	Conexiune obișnuită

O *conexiune de încredere implicită* este identică cu o conexiune obișnuită, cu excepția că acordă privilegii temporare utilizatorilor, în timp ce ei folosesc conexiunea. Privilegiile de rol care sunt acordate (dacă sunt) sunt specificate în contextul de încredere care determină ca conexiunea să fie de încredere.

Conexiunile de încredere implicite pot fi create de aplicații care se conectează folosind DB Connect. Conexiunile de încredere implicite sunt făcute și utilizate în același mod în care sunt create și folosite și conexiunile regulate. Acest lucru înseamnă că nu sunt necesare schimbări de cod pentru ca aplicațiile existente să aibă un avantaj asupra conexiunilor de încredere implicite atâta timp cât aplicațiile se conectează prin DB Connect.

O *conexiune de încredere explicită* acordă privilegii de rol temporare utilizatorului în același mod cum conexiunile de încredere implicite o fac. În plus, o conexiune de încredere explicită vă permite să modificați ID-ul de autorizare la realizarea de acțiuni legate de conexiune. Schimbarea ID-ului de autorizare într-o conexiune de încredere explicită este referită ca *comutare de utilizatori*. ID-urile de autorizare la care puteți comuta și dacă un ID de autorizare dat cere o parolă la comutarea către el sunt definite ca parte a contextului de încredere care permite crearea conexiunii de încredere.

Comutarea utilizatorilor poate reduce, în mod semnificativ, depășirea partajării într-o conexiune de-a lungul mai multor utilizatori, în special, pentru numele de utilizatori care nu cer o parolă, pentru că, în acel caz serverul de bază de date nu autentifică ID-ul de autorizare. Atunci când utilizați caracteristica, totuși, trebuie să fiți foarte sigur că aplicația dumneavoastră nu permite comutarea către un ID de autorizare fără validarea și autentificarea ID-ului de autorizare. Altfel, creați o fisură în securitatea sistemului dumneavoastră.

Conexiuni de încredere explicite pot fi create și utilizatorul poate fi comutat la conectarea prin DB2 Connect folosind CLI sau JDBC, incluzând conexiuni stabilite XA. Crearea unei conexiuni de încredere explicite și comutarea utilizatorilor cere setarea unor atribute de conexiune speciale. Acest lucru înseamnă că aplicațiile existente vor avea nevoie să fie modificate pentru ca să funcționeze conexiunile de încredere explicite.

În afară de diferențele deja menționate, puteți folosi o conexiune de încredere (implicită sau explicită) în același mod în care ați folosi o conexiune obișnuită. Trebuie să fiți sigur, totuși, în mod explicit să deconectați o conexiune de încredere explicită când ați terminat cu ea, chiar dacă se află într-o stare de întrerupere sau de deconectare. Altfel, resursele folosite de conexiune pot să nu fie eliberate. Aceasta nu este o problemă cu conexiunile de încredere implicite.

Notă:

1.

Important: Comutarea utilizatorilor fără livrarea unei parole ocolește autentificarea bazei de date a serverului. Aplicația dumneavoastră trebuie să nu permită comutarea la un ID de autorizare fără o parolă, doar dacă aplicația a validat deja și a autentificat acel ID de autorizare. A face altfel creează o fisură în securitate.

2. Conexiunile de încredere explicite ar trebui să nu folosească autentificare CLIENT. Acest lucru nu se aplică la conexiunile de încredere implicite.
3. Aplicațiile care folosesc conexiuni de încredere explicite ar trebui rulate pe mașini sigure care au parolă de protecție și accesibile doar personalului autorizat. Acest lucru nu se aplică la conexiunile de încredere implicite.

Crearea și finalizarea unei conexiuni de încredere prin CLI

Dacă serverul bazei de date la care vă conectați este configurat să permită, puteți crea o conexiune de încredere explicită la conectarea prin CLI.

Această procedură presupune că nu folosiți un manager de tranzacție XA. Dacă folosiți un manager de tranzacție XA aveți nevoie doar să vă asigurați că managerul de tranzacție este configurat pentru să seteze valoarea de configurare TCTX la TRUE când apelează xa_open. Dacă acest lucru este făcut orice conexiune care poate fi o conexiune de încredere explicită va fi o conexiune de încredere explicită. Pentru a verifica dacă o conexiune este o conexiune de încredere explicită vedeți pasul 3.

- Baza de date la care vă conectați trebuie să suporte contexte de încredere.
- Un context de încredere trebuie definit care va recunoaște clientul ca fiind de încredere.
- Trebuie să știți ID-ul de autorizare al sistemului, care este specificat în contextul de încredere. ID-ul de autorizare sistem al unei conexiuni de încredere este ID-ul de autorizare furnizat serverului ca un nume de utilizator la crearea conexiunii. Pentru a fi de încredere conexiunea dumneavoastră într-un anumit context particular, ID-ul de autorizare sistem trebuie să fie cel specificat în contextul de încredere. Întrebați administratorul de securitate de un ID de autorizare de sistem valid și parola pentru acel ID.

Exemplele din aceste instrucțiuni folosesc limbajul C și presupun că conn este un pointer la un manipulator de conexiune valid, dar neconectat. Variabila rc se presupune că are tipul de date SQLRETURN.

1. În plus față de setarea oricăror atribute de conexiune pe care le-ați seta pentru o conexiune obișnuită, setați atributul de conexiune SQL_ATTR_USE_TRUSTED_CONTEXT la SQL_TRUE cu un apel al funcției SQLSetConnectAttr.

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

2. Conectați-vă la o bază de date așa cum ați face-o pentru o conexiune normală, prin apelarea funcției SQLConnect pentru instanță. Folosiți ID de autorizare sistem ca nume de utilizator și parola sa ca parolă. Asigurați-vă că verificați avertismentele și erorile, în special cele listate în Tabela 5.

Tabela 5. Erori care indică eșuarea la crearea unei conexiuni de încredere

SQLCODE	SQLSTATE	Semnificație
SQL20360W	01679	Conexiunea nu a putut fi stabilită ca o conexiune de încredere. A fost stabilită în schimb ca o conexiune obișnuită.

Dacă nu există erori sau avertismente care să vă indice diferit, atunci conexiunea este stabilită și este o conexiune de încredere explicită.

3. (Opțional) Puteți verifica dacă o conexiune stabilită este o conexiune de încredere explicită prin verificarea valorii atributului de conexiune SQL_ATTR_USE_TRUSTED_CONTEXT folosind funcția SQLGetConnectAttr. Dacă este setat la SQL_TRUE conexiunea este o conexiune de încredere explicită.
4. Trebuie să fiți atent, totuși, să deconectați conexiunea când ați terminat cu ea, chiar dacă se află într-o stare de întrerupere sau de deconectare. Dacă nu deconectați, în mod explicit, o conexiune de încredere explicită, unele resurse folosite de conexiune nu vor fi eliberate.

Notă:

1. Conexiunile de încredere explicite ar trebui să nu folosească autentificare CLIENT. Acest lucru nu se aplică la conexiunile de încredere implicite.
2. Aplicațiile care folosesc conexiuni de încredere explicite ar trebui rulate pe mașini sigure care sunt protejate de parole și accesibile doar personalului autorizat. Acest lucru nu se aplică la conexiunile de încredere implicite.

Comutarea utilizatorilor pe o conexiune de încredere prin CLI

Puteți comuta utilizatorii pe o conexiune de încredere explicită prin interfața linie de comandă (CLI). Pentru o descriere despre ce înseamnă comutarea utilizatorilor folosind o conexiune de încredere, vedeți subiectul în Legături înrudite.

- Conexiunea trebuie creată cu succes ca o conexiune de încredere explicită.
- Conexiunea de încredere explicită nu trebuie să fie într-o tranzacție.
- Contextul de încredere care permite conexiunii de încredere explicite să fie creată trebuie configurat pentru a permite comutarea la ID-ul de autorizare pe care îl veți comuta, de asemenea.

Exemplele din aceste instrucțiuni folosesc limbajul C și presupun că conn este un pointer la o conexiune de încredere explicită. Variabila rc se presupune că are tipul de date SQLRETURN. Variabila newuser se presupune că este un pointer la un șir de caractere care păstrează ID-ul de autorizare al utilizatorului la care vreți să comutați. Variabila passwd se presupune că este un pointer la un șir de caractere care conține parola pentru ID-ul de autorizare.

1. Apelați funcția `SQLSetConnectAttr` pentru a seta atributul `SQL_ATTR_TRUSTED_CONTEXT_USERID`. Setăți-l la ID-ul de autorizare la care vreți să comutați.

```
rc = SQLSetConnectAttr(
    conn,
    SQL_ATTR_TRUSTED_CONTEXT_USERID, newuser, SQL_NTS
);
//Verificați pentru erori
```

Asigurați-vă că verificați avertismentele și erorile, în special cele listate în Tabela 6.

Tabela 6. Erori care indică eșuarea la setarea unui nou ID de autorizare la comutarea utilizatorilor

SQLCODE	Semnificație
CLI0106E	Conexiunea nu a fost făcută.
CLI0197E	Conexiunea nu este una de încredere.
CLI0124E	Este o problemă cu valoarea furnizată. Verificați că nu este null sau nu este prea lungă, de exemplu.
CLI0196E	Conexiunea este implicată într-o unitate de lucru care împiedică comutarea utilizatorilor. Pentru a putea comuta utilizatori conexiunea trebuie să nu fie într-o tranzacție.

2. (Opțional, doar dacă nu contextul de încredere care a permis această conexiune de încredere necesită o parolă pentru ID-ul de autorizare la care comutați) Apelați funcția `SQLSetConnectAttr` pentru a seta atributul `SQL_ATTR_TRUSTED_CONTEXT_PASSWORD`. Setăți-l la parola pentru noul ID de autorizare.

```
rc = SQLSetConnectAttr(
    conn,
    SQL_ATTR_TRUSTED_CONTEXT_PASSWORD, passwd, SQL_NTS
);
//Verificați pentru erori
```

Asigurați-vă că verificați avertismentele și erorile, atât cele listate în Tabela 6, cât și cele menționate în Tabela 7.

Tabela 7. Erori care indică eșuare de a seta o parolă la comutarea utilizatorilor

SQLCODE	Semnificație
CLI0198E	Atributul <code>SQL_ATTR_TRUSTED_CONTEXT_USERID</code> nu a fost încă setat.

3. Continuați ca și cum ați lucra cu o conexiune obișnuită. Dacă folosiți un manager de tranzacție XA, comutarea utilizatorului este încercată ca parte a noii cereri, altfel, comutarea utilizatorului este încercată chiar înainte de inițializarea apelării următoarei funcții care accesează baza de date (de exemplu `SQLExecDirect`). În ambele cazuri, în plus față de erorile și avertismentele pe care le verificați de obicei, asigurați-vă că verificați erorile listate în Tabela 8 la pagina 41. Erorile din Tabela 8 la pagina 41 indică faptul că a eșuat comutarea utilizatorului.

Tabela 8. Erori care indică eșuare la comutarea utilizatorilor

SQLCODE	Semnificație
SQL1046N	Contextul de încredere care permite conexiunea de încredere nu este configurat pentru a permite comutarea la ID-ul de autorizare la care încercați să comutați. Nu veți putea comuta la acel ID de autorizare până când nu este modificat contextul de încredere.
SQL30082N	Parola furnizată nu este corectă pentru ID-ul de autorizare pe care doriți să comutați.
SQL0969N cu o eroare nativă de -20361	Există o constrângere de nivel al bazei de date care vă împiedică să comutați la utilizator.

Dacă comutarea utilizatorului eșuează, conexiunea se va afla într-o stare necunoscută până când veți comuta cu succes la un alt utilizator. Puteți comuta utilizatorii pe o conexiune de încredere într-o stare de neconectare, dar nu puteți accesa serverul de bază de date cu ea. O conexiune într-o stare de neconectare va rămâne în acea stare până când veți comuta cu succes utilizatorii pe ea.

Notă:

1. **Important:** Comutând utilizatorii fără să livrați o parolă ocolește autentificarea la serverul de bază de date. Aplicația dumneavoastră trebuie să nu permită comutarea la un ID de autorizare fără o parolă, doar dacă aplicația a validat deja și a autentificat acel ID de autorizare. A face altfel creează o fisură în securitate.
2. Specificarea unei valori NULL pentru atributul SQL_ATTR_TRUSTED_CONTEXT_USERID este echivalent cu specificarea ID-ului de autorizare sistem al contextului de încredere (ID-ul de utilizator folosit când a fost creată conexiunea de încredere explicită).
3. Când ați setat cu succes valoarea atributului de conexiune SQL_ATTR_TRUSTED_CONTEXT_USERID la o conexiune de încredere explicită, conexiunea este resetată imediat. Rezultatul resetării este ca și cum o nouă conexiune a fost creată folosind atributele de conexiune originale pentru respectiva conexiune. Această resetare se întâmplă chiar dacă valoarea pe care la care setați atributul de conexiune este ID-ul de autorizare al sistemului sau NULL sau aceeași valoare pe care o reține momentan atributul.
4. Dacă atributul SQL_ATTR_TRUSTED_CONTEXT_PASSWORD este setat, parola va fi autentificată în timpul procesării comutării utilizatorului, chiar dacă contextul de încredere, care a permis conexiunea de încredere, nu cere autentificarea la o comutare de utilizator pentru acel ID de autorizare. Aceasta rezultă într-o regie nenecesară. Această regulă nu se aplică ID-ului de autorizare al contextului de încredere. Dacă ID-ul de autorizare sistemul al contextului de încredere nu cere autentificare atunci când comutați la el, atunci el nu este autentificat chiar dacă este furnizată o parolă.

Considerente privind autentificarea DB2 Connect

Ca administrator DB2 Connect, în cooperare cu administratorul bazei de date System z sau IBM Power Systems, puteți determina unde sunt validate numele de utilizator și parolele:

- La client
- La serverul System z sau IBM Power Systems
- Logare unică și validare printr-un sistem terță parte (Kerberos).

Notă: În cazul în care clientul la distanță nu specifică nici un tip de autentificare, clientul se va lega implicit la SERVER_ENCRYPT. Dacă acest tip nu este acceptat de către server,

clientul va reîncerca utilizând valoarea corespunzătoare returnată de server. Pentru a ajuta la optimizarea performanței, specificați întotdeauna tipul de autentificare la client, pentru a evita acest flux suplimentar în rețea.

Începând cu DB2 Connect Versiunea 8.2.2 (echivalentă cu Versiunea 8.1 FixPak 9) gateway-ul nu mai este un participant pasiv în timpul negocierii autentificării. Gateway-ul are un rol activ. Tipul de autentificare specificat în intrarea directorului bazei de date la gateway înlocuiește tipul de autentificare catalogat la client. Clientul, gateway-ul și serverul trebuie să specifice, toate, tipuri compatibile. Dacă tipul de autentificare catalogat la gateway nu a fost specificat în intrarea directorului bazei de date, tipul implicit cerut de server va fi autentificarea SERVER. Dar, dacă serverul nu suportă autentificarea SERVER, se va realiza negocierea între client și server. Acest comportament se deosebește de cel al clientului, care folosește în mod implicit SERVER_ENCRYPT dacă nu a fost specificat un tip de autentificare.

Tipul de autentificare catalogat la gateway nu este folosit dacă la client a fost setată opțiunea DB2NODE sau SQL_CONNECT_NODE a API-ului Set Client. Și în aceste cazuri negocierea se face numai între client și server.

Cu DB2 Connect sunt permise următoarele tipuri de autentificare:

CLIENT

Numele de utilizator și parola sunt validate la client.

DATA_ENCRYPT

Oferă capacitatea de a cripta datele de utilizator în timpul comunicațiilor client/server. Acest tip de autentificare nu este suportat pe serverul de bază de date IBM Power Systems.

KERBEROS

Activează clientul pentru a se loga la server utilizând autentificarea Kerberos în locul combinației tradiționale, alcătuite din ID și parolă. Acest tip de autentificare necesită ca atât serverul, cât și clientul să fie activați pentru Kerberos.

SERVER

Numele utilizator și parola sunt validate la baza de date server System z sau IBM Power Systems.

SERVER_ENCRYPT

Pentru autentificarea SERVER, numele de utilizator și parola sunt validate la serverul de bază de date System z sau IBM Power Systems, dar ID-urile de utilizator și parolele transferate sunt criptate la client.

SERVER_ENCRYPT_AES

ID-urile de utilizator și parolele transferate sunt criptate folosind un algoritm de criptare Advanced Encryption Standard (AES) la client și validate la serverul de bază de date System z.

Autentificarea Kerberos este unică prin aceea că un client nu transmite ID-ul de utilizator și parola direct la server. În loc, Kerberos lucrează ca un mecanism de autentificare terță parte. Utilizatorul introduce o singură dată un ID și parola la terminalul client, iar Kerberos validează această semnare. După aceasta, Kerberos transmite automat și sigur autorizarea utilizatorului oricărei cereri de servicii locale și de rețea. Aceasta înseamnă că utilizatorul nu trebuie să reintroducă un ID și o parolă pentru a se loga pe un server DB2 la distanță. Capabilitatea de semnare unică furnizată de autentificarea Kerberos necesită ca atât DB2 Connect, cât și serverul de bază de date care se conectează să asigure suport Kerberos.

Notă: Nu există suport pentru tipul de autentificare GSSPLUGIN.

Suportul Kerberos

Nivelul de autentificare Kerberos care manipulează sistemul de tichete este integrat în mecanismul Windows 2000 Active Directory. Părțile client și server ale unei aplicații comunică cu modulele de client, respectiv, server ale furnizorului Kerberos SSP (Security Support Provider). Interfața SSPI (Security Support Provider Interface) furnizează o interfață de nivel înalt pentru Kerberos SSP și pentru alte protocoale de securitate.

Setare tipică

Pentru a configura DB2 cu autentificare Kerberos, setați:

- O politică de autorizare pentru DB2 (ca service) în directorul activ care este partajat într-o rețea și
- O relație de încredere între KDC-urile (Key Distribution Centers) Kerberos

În cel mai simplu scenariu, există cel puțin o relație de încredere KDC de configurat, adică, cea dintre KDC-ul care controlează stația de lucru client și IBM Power Systems, sau System z. OS/390 Versiunea 2 Ediția 10 sau z/OS Versiunea 1 Ediția 2 furnizează procesarea tichetului Kerberos prin facilitatea sa RACF care permite gazdei să acționeze ca un UNIX KDC.

DB2 Connect furnizează ca de obicei funcționalitatea de ruter în setarea pe 3 niveluri (3-tier). Acesta nu își asumă nici un rol în autentificare atunci când se utilizează securitatea Kerberos. Nu face decât să transmită jetonul de securitate al clientului către DB2 for IBM i sau către DB2 for z/OS. Nu este nevoie ca gateway-ul DB2 Connect să fie membru al regiunii client sau gazdă Kerberos.

Compatibilitatea cu versiunile anterioare

Cerințe minime pentru suport Kerberos DB2:

IBM Data Server:

Versiunea 8

DB2 Connect:

Versiunea 8

DB2 for z/OS:

Versiunea 7

Sugestii și indicii privind securitatea z/OS

Aceste subiect furnizează unele indicații și sugestii despre securitate pentru conectarea DB2 Connect la un server bază de date DB2 for z/OS.

Câmp extins de securitate

Asigurați-vă că DB2 for z/OS Extended Security Field este setat la YES. Acest câmp apare în panoul DB2 for z/OS, DSNTIPR.

Coduri extinse de securitate

Până la DB2 for z/OS versiunea 5.1, cererile de conectare care furnizează ID-uri utilizator sau parole pot eșua cu SQL30082 cod motiv 0, dar nici o altă indicație despre ce ar putea funcționa greșit.

DB2 for z/OS versiunea 5.1 introduce o îmbunătățire care furnizează suport pentru codurile extinse de securitate. Specificând securitate extinsă, se vor furniza diagnostice suplimentare, cum ar fi (PASSWORD EXPIRED) în plus la codul motiv.

Pentru a exploata sugestiile, parametrul de instalare DB2 for z/OS ZPARM pentru securitate extinsă ar trebui setat la valoarea YES. Folosiți DB2 for z/OS panoul de instalare DSN6SYSP pentru a seta EXTSEC=YES. Puteți de asemenea folosi panoul 1 DDF (DSNTIPR) pentru a seta acest lucru. Valoarea implicită este EXTSEC=NO. În cazul unei parole expirate, Windows, Linux, UNIX și aplicații web care folosesc DB2 Connect vor primi un mesaj de eroare SQL30082.

Securitatea TCP/IP deja verificată

Dacă doriți să furnizați suport pentru opțiunea de securitate DB2 AUTHENTICATION=CLIENT, folosiți panoul de instalare DB2 for z/OS DSNTIP4 (DDF panoul 2) pentru a seta la YES securitatea TCP/IP deja verificată.

Securitatea aplicațiilor Desktop ODBC și Java

Aplicațiile Workstation ODBC și Java folosesc SQL dinamic. Aceasta ar putea crea probleme de securitate în unele instalări. DB2 for z/OS introduce o nouă opțiune de legare DYNAMICRULES(BIND) ce permite execuția de SQL dinamic sub autorizația fie a proprietarului, fie a binder-ului.

DB2 și DB2 Connect furnizează un nou parametru de configurare CLI/ODBC CURRENTPACKAGESET în fișierul de configurare DB2CLI.INI. Acesta ar trebui setat la un nume de schemă care are privilegiile corespunzătoare. O instrucțiune SQL schema SET CURRENT PACKAGESET va fi asumată automat după fiecare conectare pentru aplicație.

Folosiți ODBC Manager pentru a actualiza DB2CLI.INI.

Suportul de schimbare a parolei

Dacă parola unui ID utilizator a expirat, o instrucțiune SQL CONNECT întoarce un mesaj de eroare, cum ar fi SQLCODE -30082 cod motiv 1. Cu DB2 Connect este posibil să modificați parola de la distanță. Prin DRDA, DB2 for z/OS poate modifica parola pentru dumneavoastră, emițând următoarea instrucțiune CONNECT:

```
CONNECT TO <database> USER <userid> USING <password>  
NEW <new_password> CONFIRM <new_password>
```

Dialogul "Change password" din DB2 Configuration Assistant poate de asemenea fi folosit pentru a modifica parola.

Tipurile de autentificare suportate cu DB2 Connect

Acest subiect listează diversele combinații de setări de autentificare și securitate care sunt suportate cu DB2 Connect.

Tipuri de autentificare pentru conexiuni TCP/IP

Protocolul de comunicație TCP/IP nu suportă opțiuni de autentificare la nivelul protocolului de rețea. Tipul de autentificare determină unde are loc autentificarea. Doar combinațiile arătate în această tabelă sunt suportate de DB2 Connect. Setarea de autentificare se află în intrarea directorului bazei de date de pe serverul DB2 Connect.

Tabela 9. Scenarii valide de autentificare

Scenariu	Setare autentificare	Validare
1	CLIENT	Client
2	SERVER	Server bază de date mainframe IBM
3	SERVER_ENCRYPT	Server bază de date mainframe IBM
4	KERBEROS	Securitate Kerberos
5	DATA_ENCRYPT	Gazdă
6	SERVER_ENCRYPT_AES	Server de bază de date gazdă

Discuție asupra tipurilor de autentificare

Următoarea discuție este valabilă pentru conexiunile descrise mai sus și sunt listate în Tabela 9. În continuare, fiecare scenariu este descris mai detaliat:

- În scenariul 1, numele de utilizator și parola sunt validate numai la clientul de la distanță. Pentru un client local, numele utilizatorului și parola sunt validate numai la serverul DB2 Connect.
Se așteaptă ca utilizatorul să fie autentificat la locația la care se înscriu. ID-ul de utilizatorului este trimis prin rețea, dar nu și parola. Utilizați acest tip de securitate numai dacă toți clienții stației de lucru au facilități adecvate care pot fi de încredere.
- În scenariul 2, numele utilizator și parola sunt validate doar la serverul bază de date mainframe IBM. ID-ul de utilizator și parola sunt trimise prin rețea de pe clientul la distanță la serverul DB2 Connect și de pe serverul DB2 Connect la serverul de bază de date mainframe IBM.
- Scenariul 3 este la fel ca scenariul 2, cu excepția că ID-ul de utilizator și parola sunt criptate.
- În scenariul 4, clientul obține un tichet Kerberos de la Kerberos KDC. Tichetul este pasat nealterat prin DB2 Connect la server, unde este validat de către server.
- Scenariul 5 este același cu scenariul 3, cu excepția că datele utilizator sunt de asemenea criptate, iar DATA_ENCRYPT nu suportă serverul de bază de date IBM Power Systems.
- Scenariul 6 este la fel ca scenariul 3, cu excepția că se folosește algoritmul de criptare AES (Advanced Encryption Standard).

Capitolul 6. Legarea aplicațiilor și utilităților (DB2 Connect)

Programele de aplicație dezvoltate utilizând SQL încorporat trebuie să fie legate la fiecare bază de date cu care vor opera. Pe platformele unde sunt disponibile aceste funcții, puteți face aceasta utilizând Command Center și Configuration Assistant.

Legarea ar trebui realizată o singură dată per aplicație, pentru fiecare bază de date. În timpul procesului de legare, planurile de acces al bazei de date sunt memorate pentru fiecare instrucțiune SQL care va fi executată. Aceste planuri de acces sunt livrate de dezvoltatorii de aplicații și sunt conținute în *fișierele de legare* care sunt create în timpul precompilării. Legarea este un proces de procesare a acestor fișiere de legare de un server bază de date mainframe IBM.

Deoarece câteva dintre utilitățile furnizate cu DB2 Connect sunt dezvoltate folosind SQL înglobat, ele trebuie să fie legate la serverul de bază de date mainframe IBM înainte de a fi folosite cu sistemul respectiv. Dacă nu folosiți utilitățile și interfețele DB2 Connect, nu trebuie să le legați la fiecare dintre serverele de bază de date mainframe IBM. Listele fișierelor de legătură necesare acestor utilitare sunt conținute în următoarele fișiere:

- ddcsmvs.lst pentru System z
- ddcsvse.lst pentru VSE
- ddcsvm.lst pentru VM
- ddcs400.lst pentru IBM Power Systems

Legarea uneia sau a mai multor astfel de liste de fișiere la o bază de date va lega fiecare utilitar individual la baza de date.

Dacă este instalat un produs server DB2 Connect, utilitățile DB2 Connect trebuie să fie legate la fiecare server de bază de date mainframe IBM înainte de a fi folosite cu sistemul respectiv. Presupunând că aveți clienții la același nivel de pachet de corecții, trebuie să legați utilitățile o singură dată, indiferent de numărul de platforme client implicate.

De exemplu, dacă aveți 10 clienți Windows și 10 clienți AIX care se conectează la DB2 for z/OS prin DB2 Connect Enterprise Edition pe un server Windows, faceți una din următoarele:

- Legați ddcsmvs.lst de la unul din clienții Windows.
- Legați ddcsmvs.lst de la unul din clienții AIX.
- Legați ddcsmvs.lst dev la serverul DB2 Connect.

Acest exemplu presupune că:

- Toți clienții sunt la același nivel de service. Dacă nu este așa atunci, în plus, se poate să fie nevoie legați de la fiecare client al unui anumit nivel de service
- Serverul este la același nivel de service ca și clienții. Dacă nu este, atunci trebuie să legați și de la server.

Pe lângă utilitățile DB2 Connect, orice altă aplicație care utilizează SQL încorporat trebuie de asemenea să se lege la fiecare bază de date cu care doriți ca ele să lucreze. O aplicație care nu este legată va produce în mod normal un mesaj de eroare SQL0805N la execuție. S-ar putea să doriți să creați un fișier listă de legături adițional pentru toate aplicațiile dumneavoastră care trebuie legate.

Pentru fiecare server bază de date mainframe IBM la care legați, faceți următoarele:

1. Asigurați-vă că aveți autorizare suficientă pentru sistemul de gestiune a serverului bază de date mainframe IBM:

System z

Cerințele de autoritate sunt:

- SYSADM sau
- SYSCTRL sau
- BINDADD și CREATE IN COLLECTION NULLID

Notă: Privilegiile BINDADD și CREATE IN COLLECTION NULLID furnizează autoritate suficientă **doar** când pachetele nu există deja. De exemplu, dacă le creați pentru prime dată.

Dacă pachetele există deja și le legați din nou, atunci autoritatea necesară pentru a completa taskul (taskurile) depinde de cine a rezolvat legătura originală.

A) Dacă ați făcut legarea originală și faceți legarea din nou, atunci având oricare din autorizările listate mai sus vă va permite să realizați legarea.

B) Dacă legarea dumneavoastră originală a fost făcută de altcineva și dumneavoastră faceți a doua legare, atunci aveți nevoie de autorizările SYSADM sau SYSCTRL pentru a finaliza legarea. Având doar autorizările de BINDADD și CREATE IN COLLECTION NULLID nu vă permite să completați legătura. Tot este posibil să creați un pachet dacă nu aveți privilegiile SYSADM sau SYSCTRL. În această situație veți avea nevoie de privilegiul de BIND la fiecare din pachetele existente pe care intenționați să le înlocuiți.

VSE sau VM

Autorizarea necesară este autoritatea DBA. dacă doriți să utilizați opțiunea GRANT la comanda de legătură (pentru a evita acordarea accesului la fiecare pachet DB2 Connect individual), ID-ul de utilizator NULLID trebuie să aibă autorizarea să acorde autoritatea celorlalți utilizatori la următoarele tabele:

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

Pe sistemul VSE sau VM, puteți lansa:

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

IBM Power Systems

Autoritatea *CHANGE sau mai mult pentru colecția NULLID.

2. Lansați comenzi similare cu următoarele:

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


Unde *DBALIAS*, *USERID* și *PASSWORD* corespund serverului de bază de date mainframe IBM, *ddcsmvs.lst* este fișierul cu lista de legături pentru z/OS, iar *cale* reprezintă locația fișierului cu lista de legături.

De exemplu *drive:\sqllib\bnd* se aplică tuturor sistemelor de operare Windows și *INSTHOME/sqllib/bnd/* se aplică tuturor sistemelor de operare Linux și UNIX, unde *drive* reprezintă unitatea virtuală unde a fost instalat DB2 Connect și *INSTHOME* reprezintă directorul home al instanței DB2 Connect.

Puteți utiliza opțiunea *grant* a comenzii *bind* pentru a acorda privilegiul EXECUTE la PUBLIC sau unui nume de utilizator sau ID de grup specificat. Dacă nu utilizați opțiunea *grant* a comenzii *bind*, trebuie să rulați individual GRANT EXECUTE (RUN).

Pentru a găsi numele pachetelor pentru fișierele de legătură, introduceți comanda:

```
ddcspkgn @bindfile.lst
```

De exemplu:

```
ddcspkgn @ddcsmvs.lst
```

s-ar putea să producă ieșirea următoare:

Bind File	Package Name
f:\sql1lib\bnd\db2ajgrt.bnd	SQLAB6D3

Pentru a determina aceste valori pentru DB2 Connect executați utilitarul *ddcspkgn*, de exemplu:

```
ddcspkgn @ddcsmvs.lst
```

Opțional, acest utilitar poate fi utilizat pentru a determina numele pachetelor ale fișierelor de legătură individuale, de exemplu:

```
ddcspkgn bindfile.bnd
```

Notă:

- a. Folosirea opțiunii de legare *sqlerror continue* este necesară; totuși, această opțiune este specificată automat pentru dumneavoastră când legați aplicații folosind unelele DB2 sau CLP. Specificând această opțiune, erorile de legătură sunt transformate în avertismente, în așa fel încât fișierul de legătură care conține erori poate rezulta în continuare din crearea unor pachete. În schimb, aceasta permite folosirea unui fișier de legare pentru mai multe servere chiar și când implementarea unui anumit server ar putea ar putea face ca sintaxa SQL să fie invalidă. Din acest motiv, la legarea unui fișier cu listă de legături *ddcsxxx.lst* la un anumit server de bază de date mainframe IBM este de așteptat să apară unele avertismente.
 - b. Dacă vă conectați la o bază de date DB2 prin DB2 Connect, folosiți lista de legare *db2ubind.lst* și nu specificați *sqlerror continue*, care este validă numai pentru conectarea la un server de bază de date mainframe IBM. De asemenea, pentru a vă conecta la o bază de date DB2, este recomandat să folosiți clienții DB2 furnizați cu DB2 și nu DB2 Connect.
3. Utilizați instrucțiuni similare pentru a lega fiecare aplicație sau listă de aplicații.
 4. Dacă aveți clienți la distanță de la o ediție anterioară a DB2, ar putea fi nevoie să legați utilitarele de clienții de DB2 Connect.

Capitolul 7. Actualizările multilocație

Actualizarea multilocație, numită și unitate de lucru distribuită (DUOW) sau comitere în două faze, este o funcție care permite aplicațiilor dumneavoastră să actualizeze date pe mai multe servere de bază de date la distanță, cu integritatea garantată. Un exemplu poate fi o tranzacție bancară care implică un transfer de bani dintr-un cont în altul, pe un server de bază de date diferit.

Într-o asemenea tranzacție, este esențial ca actualizările care implementează operațiile de debit într-un cont să nu fie comise decât dacă sunt comise și actualizările necesare pentru a procesa creditele în celălalt cont. Considerentele de actualizare multilocație se aplică atunci când datele ce reprezintă aceste conturi sunt gestionate de două servere diferite de bază de date.

Produsele DB2 oferă un suport cuprinzător pentru actualizări multilocație. Acest suport este disponibil pentru aplicațiile dezvoltate folosind SQL-ul obișnuit, precum și pentru aplicațiile care folosesc monitoarele de procesare a tranzacțiilor (monitoare TP) ce implementează specificațiile interfeței X/Open XA. Printre exemplele de asemenea produse monitoare TP se numără IBM TxSeries CICS, IBM Message and Queuing Series, IBM Component Broker Series, IBM San Francisco Project, precum și Microsoft Transaction Server (MTS), BEA Tuxedo și multe altele. Cerințele de setare sunt diferite, în funcție de folosirea actualizării multilocație cu SQL nativ sau cu monitor TP.

Atât programele de actualizare multilocație cu SQL nativ, cât și cele cu monitor TP trebuie să fie precompilate cu opțiunile `CONNECT 2 SYNCPOINT TWOPHASE`. Ambele pot folosi instrucțiunea SQL `Connect` pentru a indica baza de date pe care vor să o folosească pentru instrucțiunile SQL care urmează. Dacă nu este nici un monitor TP care să comunice DB2 că va coordona tranzacția (precum este indicat de DB2 care primește apelurile `xa_open` de la monitorul TP pentru a stabili o conexiune bază de date), atunci software-ul DB2 va fi folosit pentru a coordona tranzacția.

La folosirea actualizării multilocație a monitorului TP, aplicația trebuie să ceară comitere sau derulare înapoi folosind API-ul monitorului TP, de exemplu `CICS SYNCPOINT`, `MTS SetAbort()`. Când se folosește actualizarea multilocație cu SQL nativ, trebuie folosite instrucțiunile obișnuite `SQL COMMIT` și `ROLLBACK`.

Actualizarea multilocație a monitorului TP poate coordona o tranzacție care accesează și manageri de resurse DB2 și non-DB2 cum ar fi Oracle, Informix sau SQLServer. Actualizarea multilocație nativă SQL este folosită doar cu servere DB2.

Pentru ca o tranzacție actualizare multilocație să funcționeze, fiecare din bazele de date care participă într-o tranzacție distribuită trebuie să fie capabile să suporte o unitate distribuită de lucru (DUOW). Momentan, următoarele servere DB2 au furnizat suport DUOW care le-a permis să participe în tranzacții distribuite:

- DB2 pentru Linux, UNIX și Windows Versiunea 8 sau mai recentă
- DB2 for z/OS Versiunea 7 sau ulterioară
- DB2 for IBM i

O tranzacție distribuită poate actualiza orice baze de date de servere mixate sau suportate. De exemplu, aplicația dumneavoastră poate actualiza mai multe tabele dintr-o bază de date DB2 pe Windows, o bază de date DB2 for z/OS și o bază de date DB2 for i, toate într-o singură tranzacție.

Activarea actualizărilor multilocație folosind Control Center

Puteți folosi Control Center pentru a asigura actualizări multilocație.

Pentru a activa actualizările multilocație:

1. Lansarea Control Center.
2. Faceți clic pe semnul [+] pentru a extinde vizualizarea arbore.
3. Cu butonul din dreapta al mouse-ului, selectați instanța pe care o configurați. Se deschide un meniu pop-up.
4. Selectați elementul de meniu **Multisite Update** —> **Configure**. Se deschide vrăjitorul Multisite Update.
5. Selectați **Use the TP monitor named below** și Specify a Transaction Processor (TP) monitor. Acest câmp va afișa valorile implicite pentru monitorul TP care l-ați activat. Dacă nu vreți să folosiți un monitor TP, selectați **Do Not Use a TP Monitor**.
Faceți clic pe **Next**.
6. Dacă folosiți un monitor TP, specificați setările managerului de puncte de sincronizare. Dacă nu folosiți un monitor TP, specificați baza de date a managerului de tranzacții.
7. Faceți clic pe **Finish**.

Testarea actualizărilor multilocație folosind Control Center

Puteți testa setarea actualizării multilocație folosind Control Center.

Pentru a testa actualizarea multilocație:

1. Selectați instanța cu butonul dreapta a mouse-ului și alegeți opțiunea meniu **Multisite Update** —> **Test** din meniul pop-up. Se deschide fereastra Test Multisite Update.
2. Selectați baza de date pe care doriți să o testați dintre bazele de date disponibile în caseta listă **Available**. Puteți folosi butoanele cu săgeată (> și >>) din mijloc pentru a muta selecțiile în și din caseta listă **Selected**. De asemenea, puteți să schimbați ID-ul și parola de utilizator selectate editându-le direct în caseta listă **Selected**.
3. Când ați terminat selecția, apăsați **OK**. Se deschide fereastra Multisite Update Test Result.
4. Fereastra Multisite Update Test Result arată care din bazele de date pe care le-ați selectat au trecut sau nu testul de actualizare. Fereastra va afișa coduri SQL și mesaje de eroare pentru cele care au eșuat. Faceți clic pe **Close** pentru a închide fereastra.
5. Faceți clic pe **Close** pentru a închide fereastra Test Multisite Update.

Actualizarea multilocație și managerul de puncte de sincronizare

Serverele de bază de date mainframe IBM necesită DB2 Connect pentru a participa la o tranzacție distribuită inițiată din aplicațiile web, Linux, Windows și UNIX. În plus, multe dintre scenariile actualizare multilocație care implică servere bază de date mainframe IBM necesită să fie configurată componenta SPM. Când este creată o instanță DB2, SPM DB2 este configurată automat cu setări implicite.

Necesitatea de a folosi SPM este dictată de alegerea protocolului (TCP/IP) și de folosirea monitorului TP. Următorul tabel oferă un sumar de scenarii care necesită folosirea SMP. De asemenea, tabela arată dacă este necesar DB2 Connect pentru acces la mainframe IBM de la mașinile Intel® sau UNIX. Pentru actualizări multilocație, componenta SPM a DB2 Connect este necesară dacă folosiți un monitor TP.

Tabela 10. Scenariile de actualizare multilocație care cer SPM – TCP/IP

Monitor de procesare tranzacții folosit?	Manager de puncte de sincronizare?	Produs necesar (alegeți unul)	Baza de date mainframe IBM suportată
Da	Da	Produs server DB2 Connect DB2 Enterprise Server Edition cu licența DB2 Connect aplicată	DB2 for z/OS V7 DB2 for z/OS V8 sau o versiune ulterioară
Nu	Nu	DB2 Connect Personal Edition Produs server DB2 Connect DB2 Enterprise Server Edition cu licența DB2 Connect aplicată	DB2 for z/OS V7 DB2 for z/OS V8 sau o versiune ulterioară

Notă: O tranzacție distribuită poate actualiza orice baze de date de servere mixate sau suportate. De exemplu, aplicația dumneavoastră poate actualiza mai multe tabele dintr-o bază de date DB2 pe Windows, o bază de date DB2 for z/OS și o bază de date DB2 for IBM i, toate într-o singură tranzacție.

Configurarea DB2 Connect cu un manager de tranzacții compatibil XA

Acest subiect descrie pașii de configurare necesari pentru a folosi serverele bază de date IBM Power Systems și System z în monitorul dumneavoastră TP.

Trebuie să aveți un monitor TP operațional și DB2 Connect instalat, precum și o conexiune configurată și testată la serverul de bază de date mainframe IBM.

Pentru a configura DB2 Connect să folosească servere bază de date IBM Power Systems și System z în monitorul dumneavoastră TP, realizați următorii pași:

1. Configurați monitorul TP astfel încât să poată accesa DB2 XA Switch. DB2 XA Switch livrează monitorului TP adresele API-urilor DB2 Connect XA. Fiecare monitor TP are o altă cale de a face asta.
2. Configurează monitorul TP cu șirul XA_OPEN din DB2. Fiecare monitor TP are propria cale de a face asta. Pentru informații despre cum să configurați un șir XA OPEN a lui DB2 pentru a fi folosit de către monitorul TP, vedeți documentația proprie a monitorului TP.
3. Dacă este necesar, modificați parametrii de configurare impliciți DB2 Connect SPM. Serverele de bază de date IBM, gazdă și System i (Versiunea 5 Ediția 3 și anterioare) nu suportă deocamdată interfața XA. System i Versiunea 5 Ediția 4 și următoarele oferă suport complet pentru XA.

SPM este o componentă a DB2 Connect care mapează protocolul de comitere două faze XA în protocolul de comitere două faze folosit de servere bază de date mainframe IBM. Implicit, instanța DB2 are valori predefinite pentru parametrii de configurare SPM. Cel mai semnificativ parametru este parametrul de configurare a managerului bazei de date SPM_NAME. Implicit o variantă a primelor 7 caractere ale numelui de gazdă TCP/IP.

Dacă folosiți TCP/IP să vă conectați la DB2 for z/OS, atunci nu ar trebui să modificați nici una din setările implicite. În acest caz nu este necesară nici o configurare SPM din moment ce este deja operațională.

Suportul DB2 Connect pentru tranzacții cuplate slab

Suportul din DB2 Connect pentru tranzacții cuplate slab este intenționat pentru utilizatori care implementează aplicații distribuite XA care accesează DB2 for IBM i V5R4 sau mai recentă și DB2 for z/OS Versiunea 7 sau mai recentă. Acest suport permite ramuri diferite ale aceleiași tranzacții globale pentru a partaja spațiu de blocare pe DB2 for z/OS.

Suportul pentru tranzacții cuplate slab este destinat aplicațiilor .NET și COM+.

Această caracteristică reduce fereastra în care o ramură a unei tranzacții distribuite întâlnește timeout-uri de blocare sau interblocări din cauza unei alte ramuri din aceeași tranzacție globală.

Capitolul 8. Mutarea datelor cu DB2 Connect

Dacă lucrați într-un mediu complex în care trebuie să mutați datele între sistemul bază de date gazdă și o stație de lucru, puteți folosi DB2 Connect, gateway-ul pentru transferul de date între gazdă și stația de lucru (vedeți Figura 8).

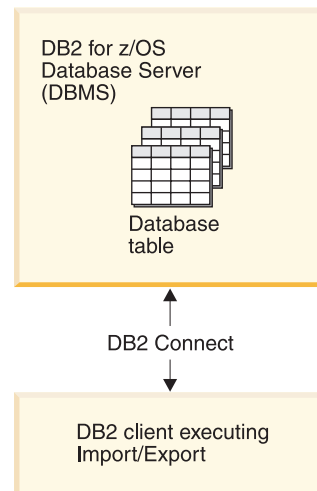


Figura 8. Import/Export prin intermediul DB2 Connect

Utilitarele de export și import DB2 vă permit să mutați date de pe un server de bază de date mainframe IBM într-un fișier de pe stația de lucru DB2 Connect și invers. Puteți folosi apoi datele cu orice alt sistem de gestionare aplicații sau baze de date relaționale care suportă acest format de export sau import. De exemplu, puteți exporta date de pe un server bază de date mainframe IBM într-un fișier PC/IXF și apoi să-l importați într-o bază de date DB2 Database for Linux, UNIX, and Windows.

Puteți realiza operații de export sau import dintr-un client de baze de date sau de pe stația de lucru DB2 Connect.

Notă:

1. Datele care vor fi exportate sau importate trebuie să se conformeze cu restricțiile legate de dimensiunea și tipul datelor care sunt aplicabile ambelor baze de date.
2. Pentru a îmbunătăți performanța, puteți folosi interogări compuse. Specificați modificatorul tipului fișierului `compus` în utilitarul de import pentru a grupa un număr specificat de instrucțiuni de interogare într-un bloc. Aceasta poate reduce regia rețelei și poate îmbunătăți timpul de răspuns.

Cu DB2 Connect, operațiile de export și import trebuie să îndeplinească următoarele condiții:

- Tipul fișierului trebuie să fie PC/IXF.
- O tabelă destinație cu atribute care sunt compatibile cu datele trebuie să fie creată pe serverul destinație înainte să puteți importa pe el. Utilitarul `db2look` poate fi folosit pentru a obține atributele tabelului sursă. Importul prin intermediul DB2 Connect nu poate crea o tabelă deoarece `INSERT` este singura opțiune suportată.

Dacă oricare din aceste condiții nu este îndeplinită, operația eșuează și este returnat un mesaj de eroare.

Notă: Definițiile index nu sunt memorate la exportare și nu sunt folosite pentru import.

Dacă exportați sau importați date amestecate (coloane ce conțin date pe un octet și pe doi octeți), considerați următoarele:

- Pe sistemele care memorează datele în EBCDIC (MVS, System z, IBM Power Systems, VM și VSE), caracterele shift-out și shift-in marchează începutul și sfârșitul datelor pe doi octeți. Când definiți lungimea coloanelor pentru tabelele bazei dumneavoastră de date, asigurați-vă că alocați suficient spațiu pentru aceste caractere.
- Sunt recomandate coloanele cu caractere de dimensiune variabilă, dacă datele din coloane nu au un model consistent.

Mutarea datelor de pe o stație de lucru pe un server gazdă

Pentru a muta datele pe un server de bază de date gazdă sau System i:

1. Exportați datele dintr-o tabelă DB2 într-un fișier PC/IXF
2. Folosind opțiunea INSERT, importați fișierul PC/IXF într-o tabelă compatibilă în baza de date a serverului gazdă.

Pentru a muta date dintr-o bază de date server gazdă pe o stație de lucru:

1. Exportați datele din tabela bazei de date server gazdă într-un fișier PC/IXF.
2. Importați fișierul PC/IXF într-o tabelă DB2.

Exemplu

Următorul exemplu ilustrează cum se mută date de pe o stație de lucru pe un server de bază de date gazdă sau System i.

Exportați datele într-un format IXF extern emițând următoarea comandă:

```
db2 export to staff.ixf of ixf select * from userid.staff
```

Emiteți comanda următoare pentru a stabili o conexiune DRDA cu baza de date DB2 destinație:

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

Dacă nu există deja, creați tabela destinație în instanța de bază de date DB2 destinație:

```
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))
```

Pentru a importa datele emiteți comanda următoare:

```
db2 import from staff.ixf of ixf insert into mydb.staff
```

Fiecare rând de date va fi citit din fișier în format IXF și o instrucțiune INSERT SQL va fi emisă pentru a insera rândul în tabela mydb.staff. Se va insera câte un rând până când sunt mutate toate datele în tabela destinație.

Puteți găsi informații detaliate în "Moving Data Across the DB2 Family," o publicație IBM Redbooks. Această publicație Redbooks poate fi găsită la următorul URL:
<http://www.redbooks.ibm.com/redbooks/SG246905>.

Capitolul 9. Maparea SQLCODE

Diferite produse bază de date relațională IBM nu produc întotdeauna aceleași coduri SQLCODE pentru erori similare. Chiar atunci când este același SQLCODE, ar putea fi însoțit de jetoane care sunt specificate diferit. Lista jetoanelor este transmisă în câmpul SQLERRMC din SQLCA. În mod implicit, DB2 Connect mapează SQLCODE-urile și jetoanele de la fiecare server de bază de date mainframe IBM la SQLCODE-urile DB2 corespunzătoare.

Dacă doriți să dezactivați maparea SQLCODE, specificați **NOMAP** în șirul de parametri ai directorului DCS.

Dacă portați o aplicație direct de pe un server de bază de date mainframe IBM, cum ar fi DB2 for z/OS, ați putea dori oprirea mapării codurilor SQLCODE. Aceasta vă va permite să folosiți aplicația fără să schimbați SQLCODE-urile pe care le referă.

Dezactivarea mapării SQLCODE

Dacă doriți să opriți maparea codurilor SQLCODE, specificați **NOMAP** în șirul de parametri ai directorului DCS.

Dacă portați o aplicație direct de pe un server de bază de date mainframe IBM, cum ar fi DB2 for z/OS, ați putea dori oprirea mapării codurilor SQLCODE. Aceasta vă va permite să folosiți aplicația fără să schimbați SQLCODE-urile pe care le referă.

Ajustarea mapării SQLCODE

În mod implicit, DB2 Connect mapează SQLCODE-urile și jetoanele de la fiecare server de bază de date mainframe IBM la SQLCODE-urile DB2 corespunzătoare. Următoarele fișiere sunt copii ale mapărilor SQLCODE implicite:

- `dcs1dsn.map` mapează SQLCODE-uri DB2 for z/OS.
- `dcs1ari.map` mapează SQLCODE-uri DB2 Server for VM and VSE.
- `dcs1qsq.map` mapează SQLCODE-uri DB2 for IBM i.

Nu sunt necesare pentru DB2 pe sisteme de operare Linux sau UNIX.

1. Dacă vreți să suprascriveți maparea implicită SQLCODE sau folosiți un server bază de date mainframe IBM care nu are o mapare SQLCODE (un server bază de date non-IBM), puteți copia unul din aceste fișiere și să îl folosiți ca bază pentru noul fișier de mapare SQLCODE. Copiind fișierul în loc să-l editați direct, vă asigurați că puteți întotdeauna face referire la maparea originală SQLCODE, dacă este necesar.
2. Specificați numele noului fișier de mapare SQLCODE în șirul parametru al directorului DCS.
3. Fiecare fișier de mapare este un fișier de mapare ASCII, care este creat și editat folosind un editor ASCII. La instalarea inițială, fișierul este memorat în directorul `mapare` din calea de instalare.

Fișierul poate conține următoarele tipuri speciale de linii:

&& Începutul logic al fișierului. Toate linii de dinainte de prima apariție a **&&** sunt considerate comentarii în formă liberă și sunt ignorate. Dacă fișierul nu conține

nimic după &&, nici o mapare SQLCODE nu este realizată. Puteți de asemenea dezactiva maparea SQLCODE cu parametrul NOMAP, așa cum s-a descris anterior.

- * Ca prim caracter într-o linie, indică un comentariu.
- W** Ca prim caracter într-o linie, indică faptul că stegulețele de avertizare ar trebui remapate. Implicit, sunt transmise stegulețele de avertizare originale. Caracterul W trebuie să fie majusculă.

Toate liniile după && trebuie să fie ori spații, ori instrucțiuni de mapare în următoarea formă:

```
cod_intrare [, cod_ieșire [, listă_jetoane]]
```

cod_intrare reprezintă una din următoarele:

sqlcode

Codul SQLCODE de la serverul bază de date mainframe IBM.

- U** Toate SQLCODE-urile nedefinite negative (cele ce nu sunt listate în acest fișier) sunt mapate la *cod_ieșire* specificat. Dacă nici un *cod_ieșire* nu este specificat în această linie, este folosit SQLCODE-ul original. Acest caracter trebuie să fie majusculă.
- P** Toate SQLCODE-urile nedefinite pozitive (cele ce nu sunt listate în acest fișier) sunt mapate la *cod_ieșire* specificat. Dacă nici un *cod_ieșire* nu este specificat în această linie, este folosit SQLCODE-ul original. Acest caracter trebuie să fie majusculă.
- ccnn** Codul de clasă SQLSTATE de la serverul bază de date mainframe IBM. *nn* este unul din următoarele:

- 00** Completare necalificată cu succes
- 01** Avertisment
- 02** Nici o dată
- 21** Încălcare de cardinalitate
- 22** Excepții date
- 23** Încălcare constrângere
- 24** Stare cursor invalidă
- 26** Identificator instrucțiuni SQL invalid
- 40** Derulare înapoi tranzacție
- 42** Încălcare acces
- 51** Stare invalidă aplicație
- 55** Obiect ce nu este în starea de cerințe preliminară
- 56** Diverse SQL sau Erori de produs
- 57** Resurse nedisponibile sau intervenție operator
- 58** Eroare sistem

Valoarea *cod_ieșire* specificată este folosită pentru toate SQLCODE-urile cu acest cod de clasă care nu sunt specificate explicit în fișierul de mapare. Dacă nici un *cod_ieșire* nu este specificat în această linie, SQLCODE-ul original este mapat la el însuși, fără să fie copiat nici un jeton peste.

Caracterele **cc** trebuie să fie litere mici.

Dacă același *cod_intrare* apare mai mult de o dată în fișierul de mapare, prima apariție este folosită. Valoarea *cod_ieșire* reprezintă SQLCODE-ul de ieșire. Dacă nu este specificată nici o valoare, SQLCODE original este folosit.

Dacă specificați un cod de ieșire, puteți de asemenea specifica un din următoarele:

- (s) SQLCODE-ul de intrare și ID-ul de produs (ARI, DSN sau QSQ) vor fi puse în câmpul jetonului de mesaje SQLCA.

SQLCODE-ul original este returnat ca fiind singurul jeton. Această opțiune este concepută pentru a trata SQLCODE-uri nedefinite, exceptând +965 și -969. Când *cod_ieșire* este +965 sau -969, lista de jetoane returnată în câmpul SQLERRMC al SQLCA include SQLCODE-ul original, urmat de identificatorul de produs, urmat de lista de jetoane originală.

Caracterul **s** trebuie să fie literă mică.

(listă-jetoane)

O listă de jetoane separate de virgule. Specificați doar o virgulă pentru a sări un anumit jeton. De exemplu, forma (*t2,,t4*) înseamnă că primul și al treilea jeton de ieșire sunt nule.

Fiecare jeton are forma unui număr (*n*), precedat opțional de **c**, urmat opțional de **c** sau **i**. Este interpretat după cum urmează:

- c** Tipul datelor jetonului în această poziție este CHAR (implicit). Dacă **c** este înainte de *n*, se referă la jetonul de intrare; dacă este după *n*, se referă la jetonul de ieșire. Caracterul **c** trebuie să fie literă mică.
- i** Tipul datelor jetonului în această poziție este INTEGER. Dacă **i** este după *n*, se referă la jetonul de ieșire. **i** nu ar trebui să fie înainte de *n*, deoarece produsele server de bază de date mainframe IBM suportă numai jetoane CHAR. Caracterul **i** trebuie să fie literă mică.
- n* Un număr de numere care indică care jetoane server bază de date mainframe IBM sunt folosite. Acestea sunt aranjate în ordinea dorită pentru plasarea în SQLCA de ieșire. Numărul indică jetonul server bază de date mainframe IBM; aranjamentul indică ordinea în care jetoanele vor fi plasate în SQLCA.

De exemplu, serverul bază de date mainframe IBM ar putea returna două jetoane, 1 și 2. Dacă vreți ca jetonul 2 să apară înainte de jetonul 1 în SQLCA de ieșire, specificați (2,1).

Pot fi combinate mai multe numere de jeton pentru a forma un jeton de ieșire CHAR, prin conectarea lor cu puncte.

Virgulele sunt folosite pentru a separa jetoanele de ieșire. Dacă nu este specificat nici un jeton înainte de o virgulă, nici un jeton de ieșire nu este inclus în SQLCA pentru această poziție. Jetoanele care apar în SQLCA de ieșire după ultimul jeton specificat sunt mapate la un jeton null.

Figura 9 la pagina 60 afișează un fișier de mapare SQLCODE eșantion.

&&			
-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)

Figura 9. Un fișier de mapare SQLCODE

Descrierile următoare corespund numărului de linie potrivit din figura anterioară:

1. SQLCODE este mapat de la -007 la -007. Primul jeton de intrare primit de la serverul bază de date mainframe IBM este folosit ca primul jeton de ieșire și valoarea sa implicită la CHAR. Nici un alt jeton nu este transferat.
2. SQLCODE este mapat de la -010 la -010 (nici un SQLCODE de ieșire nu este specificat). Nici un jeton nu este pus în SQLCA de ieșire.
3. SQLCODE este mapat de la -060 la -171. Primul jeton de intrare primit de la serverul bază de date mainframe IBM este ignorat. Al doilea este folosit ca primul jeton în SQLCA de ieșire și este CHAR. Nu există un al doilea jeton în SQLCA de ieșire.
4. SQLCODE este mapat de la -204 la -204. Primul și al doilea jeton primit de la serverul bază de date mainframe IBM sunt CHAR. Aceste două jetoane de intrare sunt combinate pentru a forma un jeton de ieșire CHAR, care va fi primul jeton de ieșire în SQLCA.
5. SQLCODE este mapat de la -633 la -206. Primul jeton de intrare primit de la serverul bază de date mainframe IBM este CHAR. Este convertit la INTEGER și este folosit ca al doilea jeton în SQLCA de ieșire. Primul jeton din SQLCA de ieșire este null, așa cum este indicat de o virgulă.
6. SQLCODE este mapat de la -30021 la -30021. Primul și al doilea jeton de intrare primit de la serverul bază de date mainframe IBM sunt CHAR și sunt folosite ca primul și al doilea jeton din ieșirea SQLCA.
7. Toate SQLCODE-urile din SQLCA-uri cu SQLSTATE din clasa 00 vor fi mapate la SQLCODE +000.
8. Toate SQLCODE-urile nedefinite sunt mapate la -969. Această opțiune trebuie să fie folosită doar dacă sunt listate toate codurile mapate, incluzându-le pe cele care sunt identice și nu necesită mapare. Opțiunea (s) indică faptul că lista de jetoane de returnat din câmpul SQLERRMC al SQLCA include SQLCODE-ul original, urmat de produsul în care a apărut eroarea, urmat de lista sa originală de jetoane. Dacă nu este inclusă intrarea U, toate codurile nelistate sunt pasate fără mapare.
9. Toate SQLCODE-urile pozitive nedefinite sunt mapate la -965. Această opțiune trebuie să fie folosită doar dacă sunt listate toate codurile mapate, incluzându-le pe cele care sunt identice și nu necesită mapare. Opțiunea (s) indică faptul că lista de jetoane de returnat din câmpul SQLERRMC al SQLCA include SQLCODE-ul original, urmat de produsul în care a apărut avertismentul, urmat de lista sa originală de jetoane. Dacă nu este inclusă intrarea P, toate codurile pozitive nelistate sunt pasate fără mapare.

Capitolul 10. Monitorizarea sistemului de bază de date și DB2 Connect

Sunt discutate câteva modalități de a monitoriza conexiunile și performanța într-un mediu de lucru în care se folosește DB2 Connect. Tipul monitorizării realizate este specific sistemului de operare.

Monitorizarea conexiunilor pentru clienți la distanță

Puteți folosi monitor de sistem bază de date cu un produs server DB2 Connect, cum ar fi DB2 Connect Enterprise Edition, pentru a monitoriza conexiunile client la distanță. Pentru a monitoriza clienții care sunt locali serverului DB2 Connect, care rulează pe însăși serverul, va trebui să setați următoarele variabile:

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

De exemplu, când apare o eroare pe sistemul mainframe IBM, administratorul de sistem poate stabili dacă problema a fost pe stația de lucru DB2 Connect. Monitorizarea sistemului de bază de date corelează:

- Jetonul de corelare DRDA (CRRTKN), pentru conversații neprotejate.
- ID-ul unității de lucru (UOWID), pentru conexiuni în două faze protejate de managerul de puncte de sincronizare DRDA-3 (când sunt utilizate peste conexiunile TCP/IP).
- Identificatorul de conexiune DB2 Connect (ID-ul aplicației).

Această informație arată care dintre conexiunile DB2 Connect a cauzat problema, ceea ce permite administratorului de sistem să forțeze aplicația client de pe sistem fără a-i afecta pe ceilalți clienți, utilizând conexiunea DB2 Connect.

Listarea stării de comutare a monitorizării

Pentru a lista starea de comutare a monitorizării, utilizați comanda `db2 get monitor switches`.

Monitorizarea performanțelor utilizând Windows Performance Monitor

Sistemele de operare Windows furnizează o unealtă utilă pentru monitorizarea performanței aplicațiilor dumneavoastră DB2. Monitorul de performanță, care este unul din uneltele administrative Windows, afișează o reprezentare grafică a performanței sistemului. Puteți alege o varietate de sisteme, baze de date și articole legate de comunicații pentru a le monitoriza și mapa împreună într-o reprezentare grafică.

De exemplu, rapoartele disponibile prin `GET SNAPSHOT FOR ALL DCS DATABASES` sau comenzile `GET SNAPSHOT FOR ALL DCS APPLICATIONS` pot fi realizate grafic în timp real utilizând monitorizarea și comparate direct cu valori cum ar fi utilizarea CPU. Puteți compara direct efectele diferitelor setări de baze de date sau performanțe în comunicație. Puteți salva configurațiile dumneavoastră specializate de setări în fișiere PMC pe care mai târziu le puteți extrage.

De exemplu în figura de mai jos, mai multe măsuri DB2 sunt afișate pe folosirea CPU. Colecția de valori care este realizată grafic a fost salvată în fișierul `db2chart.pmc`. Puteți salva câte fișiere PMC doriți, fiecare reflectând altă secțiune a performanței sistemului.

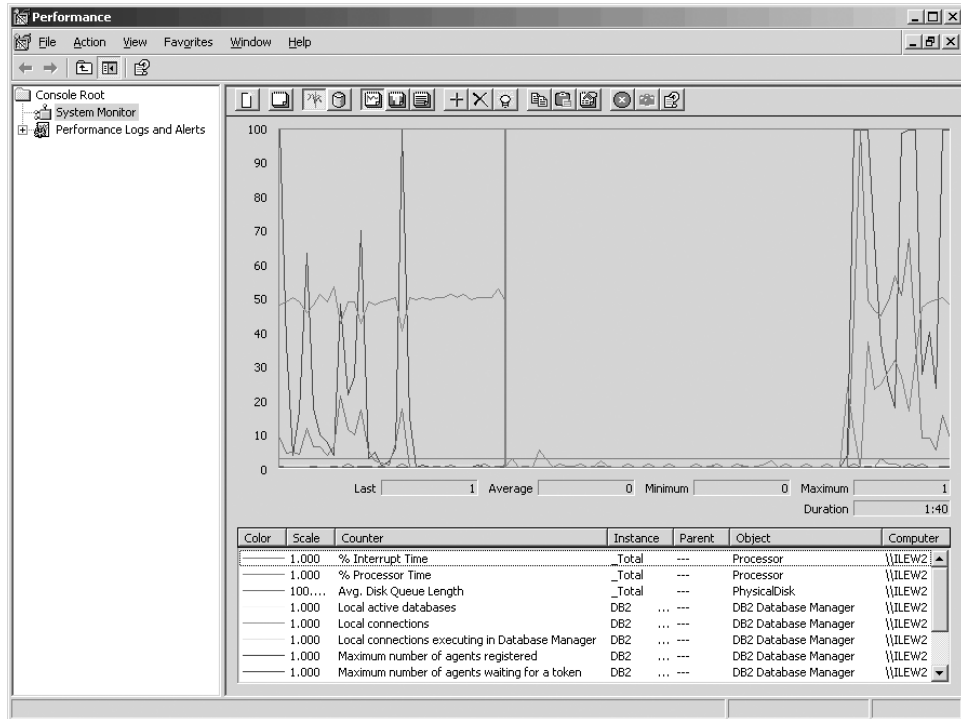


Figura 10. Performance Monitor

Pentru a activa monitorizarea aplicațiilor locale va trebui să opriți variabila de mediu DB2CONNECT_IN_APP_PROCESS.

Utilizarea comenzilor GET SNAPSHOT

Monitorul DB2 ține evidența informațiilor de sistem importante. Puteți obține un sumar al stării sistemului în orice moment lansând comanda GET SNAPSHOT. Puteți face instanțee ale monitorului dacă aveți autorizarea SYSMAINT, SYSCTRL sau SYSADM pentru instanța de monitor a sistemului de bază de date pe care doriți să o monitorizați.

Există cinci comenzi de instanțee utile pentru monitorizarea informațiilor DCS. Acestea sunt:

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

Fiecare comandă de instanțee va produce un raport detaliat despre zona pe care ați cerut-o.

De exemplu, lansarea GET SNAPSHOT FOR DCS DATABASE ON DCSDB va produce următorul raport:

DCS Database Snapshot

```
DCS database name           = DCSDB
Host database name          = GILROY
First database connect timestamp = 12-15-2001 10:28:24.596495
Most recent elapsed time to connect = 0.950561
Most recent elapsed connection duration = 0.000000
Host response time (sec.ms) = 0.000000
Last reset timestamp        =
```

```

Number of SQL statements attempted      = 2
Commit statements attempted             = 1
Rollback statements attempted           = 0
Failed statement operations              = 0
Total number of gateway connections     = 1
Current number of gateway connections   = 1
Gateway conn. waiting for host reply    = 0
Gateway conn. waiting for client request = 1
Gateway communication errors to host    = 0
Timestamp of last communication error    = None
High water mark for gateway connections = 1
Rows selected                           = 0
Outbound bytes sent                      = 140
Outbound bytes received                  = 103

```

Acest raport furnizează informații despre conexiunile, performanțele, erorile și transferul cererilor SQL pentru baza de date. Instantaneele monitorului DB2 pot fi mult mai detaliate, de fapt. De exemplu, dacă lanșați comanda GET SNAPSHOT FOR ALL DCS APPLICATIONS, veți primi un raport similar cu următorul:

DCS Application Snapshot

```

Client application ID                = 09150F74.B6A4.991215152824
Sequence number                      = 0001
Authorization ID                     = SMITH
Application name                      = db2bp
Application handle                    = 1
Application status                    = waiting for request
Status change time                   = 12-15-2001 10:29:06.707086
Client node                          = sys143
Client release level                  = SQL06010
Client platform                      = AIX
Client protocol                      = TCP/IP
Client codepage                      = 850
Process ID of client application      = 49074
Client login ID                      = smith
Host application ID                   = G9150F74.B6A5.991215152825
Sequence number                      = 0000
Database alias at the gateway        = MVSDB
DCS database name                    = DCSDB
Host database name                   = GILROY
Host release level                   = DSN05012
Host CCSID                           = 500

Outbound communication address        = 9.21.21.92 5021
Outbound communication protocol       = TCP/IP
Inbound communication address         = 9.21.15.116 46756
First database connect timestamp      = 12-15-2001 10:28:24.596495
Host response time (sec.ms)           = 0.000000
Time spent on gateway processing      = 0.000000
Last reset timestamp                  =
Rows selected                         = 0
Number of SQL statements attempted    = 2
Failed statement operations            = 0
Commit statements                     = 1
Rollback statements                   = 0
Inbound bytes received                = 404
Outbound bytes sent                   = 140
Outbound bytes received                = 103
Inbound bytes sent                    = 287
Number of open cursors                = 0
Application idle time                 = 1 minute and 32 seconds

UOW completion status                 =
Previous UOW completion timestamp     = 12-15-2001 10:28:25.592631
UOW start timestamp                   = 12-15-2001 10:29:06.142790
UOW stop timestamp                    =

```

```

Elapsed time of last completed uow (sec.ms)= 0.034396

Most recent operation                = Execute Immediate
Most recent operation start timestamp = 12-15-2001 10:29:06.142790
Most recent operation stop timestamp  = 12-15-2001 10:29:06.707053

Statement                            = Execute Immediate
Section number                        = 203
Application creator                   = NULLID
Package name                          = SQLC2C07
SQL compiler cost estimate in timerons = 0
SQL compiler cardinality estimate     = 0
Statement start timestamp             = 12-15-2001 10:29:06.142790
Statement stop timestamp              = 12-15-2001 10:29:06.707053
Host response time (sec.ms)           = 1.101612
Elapsed time of last completed stmt(sec.ms)= 0.564263
Rows fetched                          = 0
Time spent on gateway processing      = 0.013367
Inbound bytes received for statement  = 220
Outbound bytes sent for statement     = 130
Outbound bytes received for statement = 49
Inbound bytes sent for statement      = 27
SQL statement text:
create table t12 (col1 int, col2 char)

```

Starea aplicației DCS

System Monitor oferă trei forme ale comenzii LIST DCS APPLICATIONS, după cum urmează:

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

În ieșirea care urmează, formatul pentru Host Application ID și Client Application ID poate fi diferit, în funcție de versiunea bazei de date mainframe IBM și nivelul de suport TCP/IP.

Tabela 11. Formatul ID-ului de aplicație în funcție de versiunea gazdei și nivelul de suport TCP/IP

Scenariu	Format ID aplicație
Clienții care accesează servere de date cu suport RDB Manager Level mai mic de 7	G91A0D3A.P8BC.060306212019
Clienții care accesează servere de date cu nivelul de suport RDB Manager 8 sau mai mare peste TCP/IP v4	9.26.13.61.65289.060306213816
Clienții care accesează servere de date cu nivelul de suport RDB Manager 8 sau mai mare peste TCP/IP v6	2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741

LIST DCS APPLICATIONS

Pentru a vedea informațiile furnizate de monitor la nivel de aplicație, lansați comanda DB2 LIST DCS APPLICATIONS.

Returnează următoarele informații pentru o conexiune TCP/IP (DB2 Connect la DB2 for z/OS):

Auth Id	Application Name	Appl. Handle	Host Application Id
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

Auth.Id (ID de autorizare)

ID-ul de autorizare care a fost folosit pentru a vă loga la un server bază de date mainframe IBM. Acesta arată cine rulează aplicația.

Application Name (Nume aplicație)

Numele aplicației ce rulează la client, așa cum este cunoscut de DB2 Connect. Sunt disponibili doar primii 20 de octeți după ultimul separator de cale.

Appl. Handle (Mâner aplicație)

Agentul care se execută pe stația de lucru DB2 Connect. Puteți utiliza acest element pentru a lega informațiile monitorului sistemului de bază de date la alte informații de diagnosticare. ID-ul agentului este de asemenea cerut când se utilizează API-ul sau comanda FORCE USERS.

ID aplicație gazdă

Unul dintre următoarele:

- Jetonul de corelare DRDA (CRRTKN), pentru conversații neprotejate.
- ID-ul unității de lucru (UOWID), pentru conexiuni în două faze protejate de DRDA-3 Syncpoint Manager (utilizat peste conexiunile TCP/IP).

Identificatorul unic este generat când aplicația se conectează la serverul bază de date mainframe IBM. Puteți utiliza acest element împreună cu ID-ul de aplicație pentru a corela părțile de client și de server ale informațiilor de aplicație.

LIST DCS APPLICATIONS SHOW DETAIL

Dacă este specificat formatul de comandă DB2 LIST DCS APPLICATIONS SHOW DETAIL, sunt afișate informații suplimentare, incluzând:

Auth Id	Application Name	Appl. Handle	Client Application Id
NEWTON	db2cli.exe	37	2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224

Seq#	Client DB Alias	Client Node	Client Release	Client Codepage	Host Application Id
00001	MDB	SAYYID	SQL09000	1252	G91A0D3A.P982.060306214231

Seq#	Host DB Name	Host Release
00001	MEXICO	DSN08015

Client Application ID (ID aplicație client)

Identifică unic aplicația conectată la stația de lucru DB2 Connect. Există diferite formate pentru ID-ul de aplicație, care sunt dependente de protocolul de comunicație dintre client și stația de lucru DB2 Connect.

Această valoare vă permite să corelați conexiunile de la clienți la stația de lucru DB2 Connect și de la stația de lucru DB2 Connect la serverul de bază de date mainframe IBM.

Client Sequence no (Seq#) (Număr de ordine client)

Numărul de ordine client este numărul de ordine al tranzacției. Este utilizat pentru a ajuta la corelarea unei tranzacții împrăștiată pe diferite sisteme.

Client DB alias (Alias DB client)

Aliasul bazei de date furnizat de aplicație pentru conectarea la baza de date. Acest element poate fi utilizat pentru a identifica baza de date actuală pe care o accesează aplicația. Maparea dintre acest nume și numele bazei de date poate fi realizată utilizând directoarele bazei de date de pe nodul client și nodul server al managerului de bază de date.

Client NNAME (Node)

Identifică nodul unde se execută aplicația client. Informațiile variază în concordanță cu protocolul client care este utilizat. Pentru un client conectat prin TCP/IP, acesta este numele de gazdă.

Client Product ID (ID produs client)

Produsul și versiunea care rulează pe client. ID-urile produsului client vor fi:

- SQL07010 pentru versiunea 7.1 a produselor DB2 Universal Database și DB2 Connect și clienții lor.
- SQL08010 pentru versiunea 8.1 a produselor DB2 Universal Database și DB2 Connect și clienții lor.
- SQL08020 pentru versiunea 8.2 a produselor DB2 Universal Database și DB2 Connect și clienții lor.
- SQL09120 pentru Versiunea 9.1 a produselor DB2, produselor DB2 Connect și clienții lor.

Code Page ID (ID pagină de cod)

Identificatorul de pagină de cod la nodul unde a fost pornită aplicația monitorizată.

Puteți folosi aceste informații pentru a vă asigura că conversia de date este suportată între pagina de cod a aplicației și pagina de cod a bazei de date (sau pentru bazele de date ale serverului bază de date mainframe IBM, CCSID-ul serverului bază de date mainframe IBM).

Dacă pagina de cod a aplicației este diferită de cea sub care rulează monitorul sistemului de bază de date, acest element de pagină de cod vă poate ajuta să convertiți manual datele care au fost trimise de aplicație și afișate de monitorul sistemului de bază de date. De exemplu, îl puteți utiliza pentru ajutorul în traducerea numelui de aplicație.

Outbound Sequence No (Numărul de ordine al ieșirii)

Acesta reprezintă numărul de ordine al ieșirii. Este utilizat pentru corelarea tranzacțiilor între diferite sisteme.

Host Database Name (Numele bazei de date gazdă)

Numele real al bazei de date la care este conectată aplicația. În directorul DCS, acesta este *numele bazei de date destinație*.

Host Product ID (ID produs gazdă)

Produsul și versiunea care rulează pe server. Are forma *PPPVVRRM*, unde:

- PPP** Identifică produsul server de bază de date mainframe IBM (de exemplu DSN pentru DB2 for z/OS, ARI pentru DB2 Server for VSE & VM sau QSQ pentru DB2 for IBM i)
- VV** Reprezintă un număr de versiune de două cifre, cum ar fi 08.
- RR** Reprezintă un număr de ediție de două cifre, cum ar fi 01.
- M** Reprezintă nivelul de modificare, de un caracter (0-9 sau A-Z).

LIST DCS APPLICATIONS EXTENDED

Puteți utiliza comanda LIST DCS APPLICATIONS cu opțiunea EXTENDED pentru a genera un Raport extins. Raportul extins listează toate câmpurile care sunt apar când este specificată în comandă opțiunea SHOW DETAIL, plus nouă câmpuri noi:

- Starea aplicației DCS
- Status change time (Timp modificare stare)
- Client platform (Platformă client)
- Client protocol (Protocol client)
- CCSID (Host Coded Character Set Identifier - Identificator set de caractere codat gazdă).
- Client login ID (ID logare client)
- Process ID of client application (ID-ul de proces al aplicației client)
- Database alias at the gateway (Aliasul bazei de date la gateway)
- DCS database name (Numele bazei de date DCS)

În timp ce opțiunea existentă a comenzii listează câmpurile orizontal, câte o linie pentru aplicație, noua opțiune le listează vertical, câte un câmp pe linie.

Iată noua sintaxă a comenzii:

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

Acesta este un exemplu de ieșire a acestei comenzi, când se utilizează noua opțiune EXTENDED:

List of DCS Applications - Extended Report

```
Client application ID           = 2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224
Sequence number                = 00001
Authorization ID                = NEWTON
Trusted Authorization ID       =
Application name                = db2cli.exe
Application handle              = 37
Application status              = waiting for request
Status change time              = Not Collected
Client node                     = SAYYID
Client release level            = SQL09000
Client platform                 = NT
Client protocol                 = TCP/IP
Client codepage                 = 1252
Process ID of client application = 1192
Client login ID                 = ISAYYID
Host application ID             = G91A0D3A.P982.060306214231
Sequence number                 = 00001
Database alias at the gateway   = MDB
DCS database name               = MDB
Host database name              = MEXICO
Host release level              = DSN08015
Host CCSID                      = 1208
```

Câmpul de stare al aplicației conține una din următoarele valori:

1. **connect pending - outbound.** Aceasta înseamnă că cererea de conectare la o bază de date mainframe IBM a fost emisă și DB2 Connect așteaptă să fie stabilită conexiunea.
2. **waiting for request.** Conexiunea cu baza de date mainframe IBM a fost stabilită și DB2 Connect așteaptă pentru o instrucțiune SQL de la aplicația client
3. **waiting for reply.** Aceasta înseamnă că instrucțiunea SQL a fost trimisă la baza de date mainframe IBM.

De asemenea, timpul modificării stării este afișat în report doar dacă în timpul procesării a fost activat comutatorul System Monitor UOW. În caz contrar, va fi afișat, "Not Collected" (necolectat).

Monitorul de sănătate și alertele

Monitorul de sănătate din DB2 for z/OS evaluează periodic politicile de întreținere a obiectelor. Dacă monitorul de sănătate determină că este necesară întreținerea unui obiect, sunt create alerte de sănătate. Acțiunile realizate ca răspuns la alertele de sănătate pot fi vizualizate, lansate pentru rulare și salvate.

Privire generală asupra monitorului de sănătate DB2 for z/OS

Pe sistemele z/OS, monitorul de sănătate DB2 for z/OS este pornit ca un task pentru fiecare subsistem DB2 care trebuie să fie monitorizat sau pentru un membru dedicat al unui grup de partajare a datelor.

Monitorul de sănătate DB2 for z/OS declanșează evaluarea politicilor de întreținere a obiectelor la momentele și intervalele planificate, așa cum s-a definit în politică. Politicile de întreținere a obiectelor sunt create folosind vrăjitorul de creare a politicilor de întreținere a obiectelor, din Centrul de control DB2. În timpul fiecărei evaluări de politică, este verificat criteriul pentru recomandarea întreținerii pe baza pragurilor setate în politica de întreținere a obiectelor, pentru a determina necesitatea întreținerii obiectelor (adică dacă este nevoie de COPY, REORG, RUNSTATS, STOSPACE, ALTER TABLESPACE sau ALTER INDEX) și a identifica stările restricționate, cum ar fi CHKP, pentru obiectele din grupul de stocare, spațiul de tabelă și index, dacă este cazul. Dacă în timpul evaluării politicii sunt identificate obiecte aflate în stare de alertă, sunt notificate contactele din politică pentru alertarea cu privire la sănătate, la adresele lor de e-mail sau numerele de pager. Lista contactelor de alertat cu privire la sănătate pentru fiecare subsistem DB2 este definită și gestionată în Centrul de control.

Când este pornit, monitorul de sănătate realizează un instantaneu inițial al planificării evaluării politicilor, pe care îl folosește pentru a determina când trebuie să declanșeze evaluările de politică. Acest instantaneu de planificare este reîmprospătat la momentele specificate la pornirea monitorului de sănătate sau atunci când monitorul de sănătate primește o comandă de reîmprospătare. Modificările din planificarea de evaluare a unei politici sunt preluate de monitorul de sănătate atunci când se realizează reîmprospătarea planificării.

Monitorul de sănătate este pornit și oprit de la consolă, folosind comenzile START și STOP ale sistemului MVS.

Jobul de instalare DSNTIJHM plasează în biblioteca de proceduri un exemplu de procedură catalogată (DSNHMONP) care pornește un monitor de sănătate DB2 și un exemplu de procedură catalogată (DSNHMONA) care pornește mai multe monitoare de sănătate DB2 într-un sistem MVS sau într-un Parallel Sysplex.

Vizualizările, tabelele, seturile de date, procedurile catalogate, procedurile memorate, funcțiile definite de utilizator și tabela cu setul de rezultate folosite de monitorul de sănătate db2 sau de taskurile înrudite de mai jos sunt create și instalate de joburile de instalare DSNTIJCC și DSNTIJHM. DSNTIJCC și DSNTIJHM sunt livrate cu FMID-urile JDB771D și JDB881D.

Istoricul de evaluare a politicii

Evaluările de politică declanșate de monitorul de sănătate DB2 sunt înregistrate în tabela DSNACC.HM_EVAL_LOG. În istoric sunt create intrări atunci când începe evaluarea de politică și atunci când se termină evaluarea. Intrările istoricului sunt păstrate 7 zile, după care

sunt șterse din tabelă. Poate fi folosită vizualizarea DB2 DSNACC.HM_ALERT_PO_EV, creată pentru această tabelă de jobul de instalare DSNTIJCC, pentru a afișa toate politicile a căror ultimă evaluare nu s-a realizat cu succes.

Pornirea, oprirea și reîmprospătarea monitorului de sănătate DB2 for z/OS

Pe sistemul z/OS, monitorul de sănătate DB2 for z/OS este pornit ca un task pentru fiecare subsistem DB2 care trebuie să fie monitorizat sau pentru un membru dedicat al unui grup de partajare a datelor.

- Pentru a porni un monitor de sănătate DB2, lansați următoarea comandă START în sistemul MVS:

```
S membername, DB2SSN=ssid, JOBNAME=HMONssid, TRACE=trace, REFRESH=nn
```

Parametrii TRACE și REFRESH sunt opționali.

membername

Specifică un membru al bibliotecii de proceduri executat pentru a porni monitorul de sănătate DB2, și anume DSNHMONP. Această procedură catalogată este creată de jobul de instalare DSNTIJHM.

ssid

Specifică numele sau identificatorul subsistemului DB2 care trebuie să fie monitorizat.

trace

Specifică stegulețul de urmărire. Valorile posibile sunt:

- ON - Se activează urmărirea. Înregistrările de urmărire sunt scrise în SYSOUT
- OFF - Nu se activează urmărirea.

Valoarea implicită este OFF.

nn

Specifică ora (în formatul de 24 de ore) la care monitorul de sănătate reîmprospătează instantaneul planificării evaluării, folosit la declanșarea evaluărilor de politică. Valoarea implicită este 22.

- Pentru a porni mai multe monitoare de sănătate DB2, lansați următoarea comandă START în sistemul MVS:

```
S membername
```

membername

Un membru al bibliotecii de proceduri executat pentru a porni mai multe monitoare de sănătate DB2, și anume DSNHMONA.

Notă: Înainte de a porni mai multe monitoare de sănătate DB2 cu o singură comandă START folosind DSNHMONA, setul de date HMONPARM specificat în procedura DSNHMONA trebuie să fie populat cu lista subsistemelor care trebuie să fie monitorizate. Procedura catalogată și setul de date sunt create de jobul de instalare DSNTIJHM.

- Pentru a reîmprospăta instantaneul planificării evaluărilor de politică, folosit de monitorul de sănătate DB2 pentru a determina când trebuie să declanșeze evaluarea politicii, lansați următoarea comandă MODIFY în sistemul MVS:

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

ssid

Numele sau identificatorul subsistemului DB2 care este monitorizat de monitorul de sănătate DB2 pe care îl reîmprospătați.

- Pentru a opri un monitor de sănătate DB2, lansați următoarea comandă STOP în sistemul MVS:

STOP HMONssid sau P HMONssid
ssid

Numele sau identificatorul subsistemului DB2 care este monitorizat de monitorul de sănătate DB2 pe care îl opriți.

Vizualizarea, lansarea și salvarea acțiunilor recomandate

Pentru a vizualiza, lansa și salva acțiunile recomandate pentru obiectele de alertă identificate în timpul evaluării de politică, apelați procedura memorată DB2 SYSPROC.DSNACCHR, care este creată de jobul de instalare DSNTIJCC. DSNACCHR este o procedură memorată ce determină acțiunile recomandate pentru obiectele de alertă identificate în timpul evaluării de politică și generează un job JCL care va executa acțiunile recomandate.

Următoare diagramă de sintaxă prezintă instrucțiunea SQL CALL pentru invocarea DSNACCHR. Deoarece convenția de legare pentru DSNACCHR este GENERAL WITH NULLS, dacă transmiteți parametri în variabile gazdă trebuie să includeți un indicator null pentru fiecare variabilă gazdă. Trebuie să inițializați indicatorii null pentru variabilele de intrare gazdă înainte de a executa instrucțiunea CALL.

Sintaxa

```
▶▶CALL DSNACCHR (—query-type,—health-ind,—policy-id,—work-set,——————▶
▶dataset-name,—member-name,—save-opt,—trace-flag,——————▶
  └─NULL┘      └─NULL┘      └─NULL┘
▶—job-id,—jobname,—jcl-proc-time,—trace-flag,—last-statement,——————▶
▶—return-code,—error-msg )—————▶▶
```

query-type

Specifică ce doriți să faceți cu acțiunile recomandate pentru obiectele identificate ca fiind în stare de alertă în timpul evaluării politicii. Valorile posibile sunt:

- 0 - Vizualizarea acțiunilor recomandate pentru obiectele de alertă ca job JCL
- 1 - Lansarea jobului JCL care execută acțiunile recomandate pentru obiectele de alertă
- 2 - Lansarea jobului JCL care execută acțiunile recomandate pentru obiectele de alertă și punerea jobului în coada de așteptare
- 3 Salvarea acțiunilor recomandate pentru obiectele de alertă ca job JCL într-un membru al bibliotecii

query-type este un parametru de intrare de tip INTEGER.

health-ind

Specifică tipul de alertă pe care îl include DSNACCHR în jobul JCL. Valorile posibile sunt:

- RS - Stare restricționată
- EX - Extensii depășite
- RR - Necesită REORG
- CR - Necesită COPY
- RT - Necesită RUNSTATS
- SS - Necesită STOSPACE

health-ind este un parametru de intrare de tip VARCHAR(4).

policy-id

Specifică o politică de întreținere a obiectelor. *policy-id* este un parametru de intrare de tip VARCHAR(7).

work-set

Specifică setul de lucru al unei politici de întreținere care a identificat obiectele de alertă pe care DSNACCHR le include în jobul JCL. Acest set de lucru trebuie să fie identificat cu politica și tipul de alertă specificate de parametrii *policy-id* și *health-ind*. *work-set* este un parametru de intrare de tipul INTEGER.

dataset-name

Specifică un set de date partiționate (PDS) complet calificat sau un nume de set extins de date partiționate (PDSE). Această valoare trebuie să fie specificată dacă *query-type* este 3. *dataset-name* este un parametru de intrare de tipul VARCHAR(44).

member-name

Specifică un membru al setului de date partiționate (PDS) sau al setului extins de date partiționate (PDSE) specificat în parametrul *dataset-name*, în care va fi salvat jobul JCL de întreținere. Această valoare trebuie să fie specificată dacă *query-type* este 3. *member-name* este un parametru de intrare de tip VARCHAR(8).

save-opt

Specifică modul în care se salvează jobul JCL de întreținere a obiectelor. Această valoare trebuie să fie specificată dacă *query-type* este 3. Valorile posibile sunt:

- R - Înlocuire
- A - Adăugare la sfârșit
- NM - Membru nou

save-opt este un parametru de intrare de tip VARCHAR(2).

trace-flag

Specifică dacă urmărirea va fi activată sau dezactivată. Valorile posibile sunt:

- Y - Se activează urmărirea
- N - Se dezactivează urmărirea

trace-flag este un parametru de intrare de tip CHAR(1).

job-ID

Când *query-type* este 1 sau 2, specifică ID-ul de job al jobului lansat. *job-id* este un parametru de intrare de tip VARCHAR(8).

jobname

Când *query-type* este 1 sau 2, specifică numele jobului lansat. *jobname* este un parametru de intrare de tip VARCHAR(8).

jcl-proc-time

Specifică momentul în care a fost procesată cererea. *jcl-proc-time* este un parametru de ieșire de tip **TIMESTAMP**.

last-statement

Când **DSNACCHR** returnează o eroare gravă (codul de retur 12), acest câmp conține instrucțiunea SQL care a fost executată când a apărut eroarea. *last-statement* este un parametru de ieșire de tip **VARCHAR(2500)**.

return-code

Codul returnat de execuția **DSNACCHR**. Valorile posibile sunt:

- 0 - **DSNACCHR** s-a executat cu succes
- 12 - **DSNACCHR** s-a terminat cu erori grave. Parametrul *error-msg* conține un mesaj care descrie eroarea. Parametrul *last-statement* conține instrucțiunea SQL care se executa când a apărut eroarea.

return-code este un parametru de tip **INTEGER**.

error-msg

Când **DSNACCHR** returnează o eroare gravă (codul de retur 12), acest câmp conține mesaje de eroare, inclusiv **SQLCA** formatat. *error-msg* este un parametru de ieșire de tip **VARCHAR(1331)**.

DSNACCHR returnează un set de rezultate când parametrul *query-type* este 0. Setul de rezultate conține jobul JCL generat de **DSNACCHR**. Tabela cu setul de rezultate **DSNACCHR** este creat de jobul de instalare **DSNTIJCC**. Tabela 12 afișează formatul setului de rezultate.

Tabela 12. Formatul setului de rezultate DSNACCHR

Nume coloană	Tip date	Descriere
JCLSEQNO	INTEGER	Numărul de ordine al rândului din tabelă (1,...,n)
JCLSTMT	VARCHAR(80)	Specifică o instrucțiune JCL

Vizualizarea sumarelor de alerte privind sănătatea

Funcția **HEALTH_OVERVIEW** returnează informații din setul de date **KSDS VSAM Health Alert Summary** ca o tabelă **DB2**. Acest set de date este creat de jobul de instalare **DSNTIJHM**.

Setul de date **Health Alert Summary** conține informații privind starea monitorului de sănătate **DB2** și statisticile sumarelor de alerte pentru fiecare subsistem **DB2** monitorizat anterior sau curent de monitorul de sănătate în sistemul **MVS** sau **Parallel Sysplex**. Aceste informații sunt returnate clientului cu un rând pentru fiecare subsistem **DB2** și recomandare privind alerta.

Rezultatul funcției este o tabelă **DB2** cu următoarele coloane:

ip-addr

Adresa IP a serverului **DB2**. Aceasta este o coloană de tip **VARCHAR(40)**.

db2-ssid

Identificatorul subsistemului DB2. Aceasta este o coloană de tip VARCHAR(4).

health-ind

Tipul de alertă. Valorile posibile sunt:

- RS - Stare restricționată
- EX - Extensii depășite
- RR - Necesită REORG
- CR - Necesită COPY
- RT - Necesită RUNSTATS
- SS - Necesită STOSPACE
- PO - Evaluare de politică eșuată
- HM - Stare monitor de sănătate

health-ind este o coloană de tip VARCHAR(4).

host-name

Numele de domeniu complet calificat al serverului DB2. Aceasta este o coloană de tip VARCHAR(255).

summary-stats

Starea monitorului de sănătate DB2 dacă *health-ind* este 'HM'. Valorile posibile sunt:

- 0 Monitorul de sănătate nu este pornit
- 1 Monitorul de sănătate este pornit
- -1 Starea monitorului de sănătate este necunoscută

Altfel, numărul total de obiecte de alertă cu tipul de alertă specificat în *health-ind*. Aceasta este o coloană de tip INTEGER.

alert-state

Starea alertei specificate în *health-ind*. Valorile posibile sunt:

- 5 - Alarmă
- 4 - Atenție
- 3 - Avertisment
- 0 - Normal

alert-state este întotdeauna 0 când *health-ind* este 'HM'. Aceasta este o coloană de tip INTEGER.

Numele programului extern pentru funcție este HEALTH_OVERVIEW, iar numele specific este DSNACC.DSNACCHO. Această funcție este creată de jobul de instalare DSNTIJCC.

Exemplu: Găsiți numărul total de obiecte de alertă care necesită COPY pentru subsistemul DB2 'ABCD':

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

Vizualizarea obiectelor de alertă privind sănătatea

Obiectele de alertă identificate în timpul ultimei iterații realizate cu succes a evaluării de politică sunt salvate în aceste tabele depozit de obiecte de alertă, în funcție de tipul obiectului.

Obiectele de alertă sunt:

- DSNACC.HM_MAINT_TS pentru spațiile de tabelă
- DSNACC.HM_MAINT_IX pentru indecși
- DSNACC.HM_MAINT_SG pentru grupurile de stocare

DB2 creează un număr de vizualizări pentru aceste tabele depozit de obiecte de alertă. Vizualizările și tabelele depozit de obiecte de alertă sunt create de jobul de instalare DSNTIJCC. Tabela 13 listează tabelele pentru care sunt definite vizualizări și descrieri de vizualizări. Toate numele de vizualizări și numele de tabele au calificativul DSNACC.

Tabela 13. Vizualizările pentru obiectele de alert privind sănătatea

Nume vizualizare	Tabelă	Descriere vizualizare
HM_ALERT_TS_RS	HM_MAINT_TS	Afișează toate spațiile de tabelă în stare restricționată
HM_ALERT_TS_EX	HM_MAINT_TS	Afișează toate spațiile de tabelă ale căror extensii au depășit o limită specificată de utilizator
HM_ALERT_TS_RR	HM_MAINT_TS	Afișează toate spațiile de tabelă care necesită REORG
HM_ALERT_TS_CR	HM_MAINT_TS	Afișează toate spațiile de tabelă care necesită COPY
HM_ALERT_TS_RT	HM_MAINT_TS	Afișează toate spațiile de tabelă care necesită RUNSTATS
HM_ALERT_IX_RS	HM_MAINT_IX	Afișează toți indecșii care sunt în stare restricționată
HM_ALERT_IX_EX	HM_MAINT_IX	Afișează toți indecșii ale căror extensii au depășit o limită specificată de utilizator
HM_ALERT_IX_RR	HM_MAINT_IX	Afișează toți indecșii care necesită REORG
HM_ALERT_IX_CR	HM_MAINT_IX	Afișează toți indecșii care necesită COPY
HM_ALERT_IX_RT	HM_MAINT_IX	Afișează toți indecșii care necesită RUNSTATS
HM_ALERT_SG_SS	HM_MAINT_SG	Afișează toate grupurile de stocare care necesită STOSPACE

Partea 3. Disponibilitatea înaltă și DB2 Connect

Pentru un mediu de lucru în care se folosește DB2 Connect există considerente specifice în ceea ce privește disponibilitatea înaltă. Dacă, dintr-un anumit motiv, un server de bază de date dintr-o rețea devine indisponibil, este importantă capacitatea de a redirecționa o stație de lucru client la un server alternativ de bază de date.

Capitolul 11. Disponibilitatea înaltă și echilibrarea încărcării pentru conectivitatea bazei de date gazdă

Pe piața actuală a tehnologiei informației există o mare cerere pentru disponibilitatea non-stop a datelor. Această cerință trebuie îndeplinită pentru ca o afacere să țină pasul cu concurența și să își continue dezvoltarea. Multe dintre aplicațiile actuale de Web, de e-business și de calcul tabelar necesită acces la datele de întreprindere. Trebuie stabilită o conexiune de încredere, rapidă și securizată la bazele de date mainframe IBM. Această conexiune trebuie să fie disponibilă constant și să poată trata cererile de conexiune în condiții de încărcare critică. Cum poate fi construită o astfel de conexiune?

Scenariu de disponibilitate ridicată

O companie are mai multe stații de lucru și servere de aplicații care rulează pe Windows, Linux și UNIX. Aceste mașini necesită acces la date care se află pe mai multe baze de date mainframe IBM. Aplicațiile care rulează pe aceste mașini necesită conexiuni rapide și fiabile la bazele de date. Întregul sistem este conectat printr-o rețea Ethernet utilizând TCP/IP.

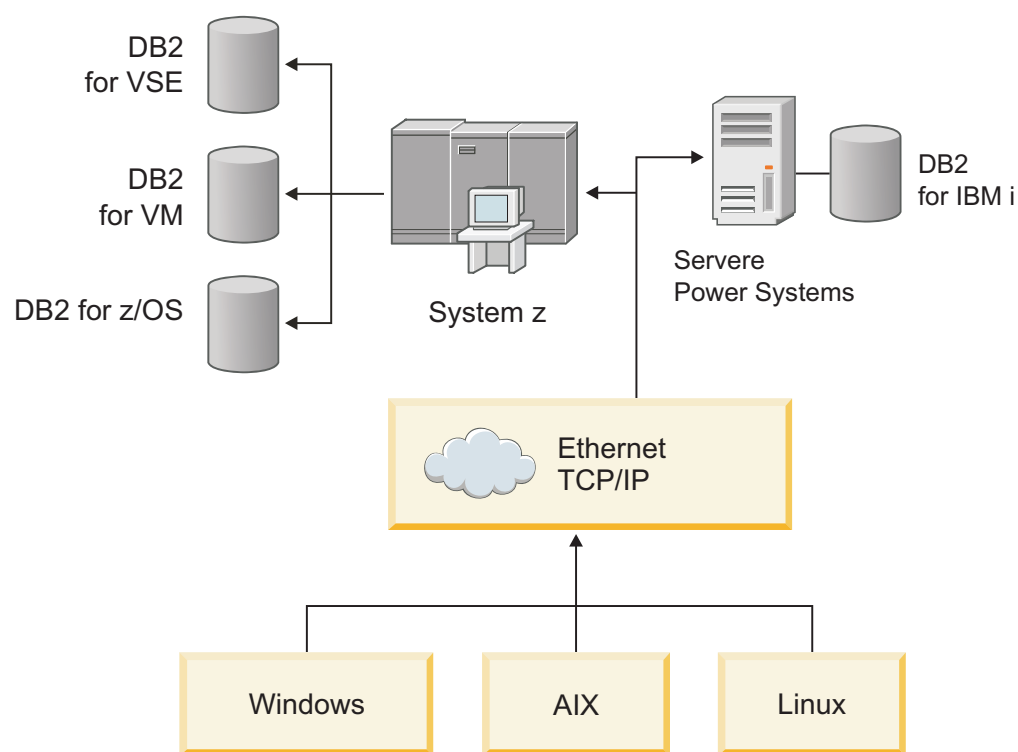


Figura 11. Scenariu de rețea exemplu

Pentru stații de lucru și servere de aplicații care accesează baze de date mainframe IBM, aveți nevoie de o componentă de conectivitate ca intermediar. Această componentă trebuie să furnizeze o conexiune la bazele de date mainframe IBM care să fie de înaltă disponibilitate, robustă și rapidă. De asemenea, trebuie să fie scalabilă, pentru a anticipa viitoarea creștere a volumului de conexiuni.

Folosiți legăturile înrudite din acest subiect pentru a vedea detalii despre soluția bazată pe DB2 Connect și caracteristica de rerutare automată a clientului.

Capitolul 12. Descrierea și setarea rerutării automate a clientului (DB2 Connect)

Principalul scop al caracteristicii de rerutare automată client este aceea de a activa o aplicație client IBM Data server pentru a recupera dintr-o pierdere de comunicații, astfel încât aplicația să poată continua să funcționeze cu un minimum de întreruperi. Așa cum arată și numele, rerutarea este elementul central al suportului pentru operații continue. Dar rerutarea este posibilă doar atunci când există o locație alternativă care este identificată conexiunii clientului. Într-un mediu de înaltă disponibilitate non-DB2 Connect, baza de date care este accesată este în mod obișnuit sincronizată între serverul DB2 original și serverul DB2 alternativ, prin unul din diversele mijloace, cum ar fi HACMP (High Availability Cluster MultiProcessor) sau HADR (High Availability Disaster Recovery).

În cazul serverului DB2 Connect, deoarece nu este nevoie de sincronizarea bazelor de date locale, trebuie doar să vă asigurați că atât serverul DB2 Connect original, cât și cel alternativ, au baza de date mainframe IBM destinație catalogată, astfel încât să fie accesibilă folosind un alias de bază de date identic.

Notă: Într-un mediu de server DB2 Connect alternativ, poate fi specificat un server DB2 Connect pentru a activa rerutarea automată între un client și serverul DB2 Connect. Pentru ca rerutarea să aibă loc între produsele DB2 Connect personal sau server și un server de bază de date mainframe IBM, serverul de la distanță trebuie să furnizeze una sau mai multe adrese alternative pentru el. În cazul DB2 for z/OS, se cunosc mai multe adrese dacă baza de date este un mediu Sysplex de partajare a datelor.

Capabilitatea de rutare pentru Sysplex poate fi configurată între DB2 Connect și serverul de bază de date gazdă, dacă este activat suportul Sysplex. Capabilitatea de rerutare pentru Sysplex este o caracteristică DB2 Connect care permite DB2 Connect să reîncece conexiunea la alți membri din grupul Sysplex, după ce s-a pierdut conexiunea la membrul original. Serverul alternativ nu este nevoie să fie catalogat în directorul bazei de date pentru a se activa capabilitatea de rerutare pentru Sysplex pe DB2 Connect. Implicit, capabilitatea de rerutare pentru Sysplex este activată, dacă suportul Sysplex este activat.

Pentru ca un client IBM Data Server să poată recupera dintr-o pierdere a comunicației cu un server DB2 Connect folosind rerutarea automată a clientului, o locație de server DB2 Connect alternativ trebuie specificată înainte de apariția pierderii de comunicație. Pentru a defini locația serverului DB2 Connect alternativ pentru o anumită bază de date mainframe IBM, este folosită comanda UPDATE ALTERNATE SERVER FOR DATABASE. Numele de gazdă alternativ și numărul de port sunt date ca parte a comenzii. Locația este memorată în fișierul director bază de date sistem pe serverul DB2 Connect. Pentru ca să fiți sigur că locația serverului DB2 Connect alternativ specificată se aplică la acea bază de date pentru toți clienții, locația serverului alternativ trebuie specificată la nivelul serverului DB2 Connect. Serverul alternativ este ignorat dacă este setat ca instanță a clientului.

De exemplu, să presupunem că o bază de date mainframe IBM este catalogată folosind aliasul de bază de date db1 la serverul DB2 Connect S1 (cu un numele de gazdă db2conn1 și numărul de port 122). Administratorul bazei de date dorește să specifice un server DB2 Connect alternativ S2 la numele de gazdă db2conn2 cu numărul de port 123. Mai jos este comanda pe care administratorul bazei de date o va rula la serverul DB2 Connect S1:

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

După ce ați specificat locația serverului DB2 Connect alternativ pentru aliasul de bază de date db1 la serverul DB2 Connect S1, informațiile despre locația serverului alternativ sunt returnate clientului IBM Data Server ca parte a procesului de conectare. În cazul în care comunicațiile între clientul IBM Data Server și serverul DB2 Connect S1 se pierd pentru un motiv oarecare (de obicei o eroare de comunicație, cum ar fi codul SQL -30081 sau codul SQL -1224), clientul IBM Data Server va încerca să se reconecteze la db1 fie prin serverul DB2 Connect original (S1), fie prin serverul DB2 Connect alternativ (S2), comutând între încercările cu fiecare server. Intervalul de timp între încercări este mic la început, apoi gradual se mărește cu fiecare încercare.

Când o conectare s-a făcut cu succes, codul SQL -30108 este returnat pentru a arăta că o conexiune de bază de date a fost restabilită după eșec de comunicație. Sunt returnate numele de gazdă sau adresa IP și numele de serviciu sau numărul de port. Clientul IBM Data server întoarce aplicației doar eroarea pentru eșecul comunicațiilor inițiale, dacă restabilirea comunicațiilor clientului cu serverul original sau cu cel alternativ nu este posibilă.

Trebuie să notați următoarele considerente privitoare la conectivitatea cu server alternativ într-un mediu de server DB2 Connect:

- Când folosiți un server DB2 Connect pentru a furniza acces la o bază de date mainframe IBM atât pentru clienții locali, cât și pentru cei la distanță, poate apărea confuzia cu privire la informațiile de conectivitate ale serverului alternativ, dintr-o intrare de director bază de date sistem. Pentru a minimiza confuzia, considerați catalogarea a două intrări în directorul bazei de date sistem pentru a reprezenta aceeași bază de date mainframe IBM. Catalogați o intrare pentru clienții la distanță și o alta pentru clienții locali.
- Orice informații SYSPLEX care sunt întoarse de la un server DB2 for z/OS destinație sunt păstrate doar în cache la serverul DB2 Connect. Numai un singur server alternativ este scris pe disc. Când există mai multe servere alternative sau mai multe servere active, informațiile sunt păstrate în memorie și se pierd când se termină procesul.

Capitolul 13. Configurarea re-rutării automate a clientului pentru tehnologia de distribuitor a conexiunii client

Tehnologiile de distribuitor sau dispecer, cum ar fi WebSphere EdgeServer, distribuie cererile de reconectare a aplicației client la un set definit de sisteme dacă serverul primar de bază de date eșuează. Dacă folosiți o tehnologie de distribuitor cu re-rutarea automată a clientului DB2, trebuie să identificați distribuitorul ca server alternativ pentru re-rutarea automată a clientului DB2.

Este posibil să folosiți o tehnologie de distribuitor într-un mediu similar următorului:

Client → tehnologie de distribuitor → (Server 1 DB2 Connect sau Server 2 DB2 Connect)
→ DB2 z/OS

unde :

- Componenta de tehnologie de distribuitor are un nume de gazdă TCP/I de **DThostname**
- Serverul 1 DB2 Connect are un nume de gazdă TCP/IP de **GWYhostname1**
- Serverul 2 DB2 Connect are un TCP/IP de **GWYhostname2**
- Serverul DB2 z/OS are numele de gazdă TCP/IP **zOShostname**

Clientul este catalogat folosind **DThostname** pentru a utiliza tehnologia de distribuitor pentru a accesa oricare dintre Serverele DB2 Connect . Tehnologia de distribuitor care intervine ia decizia să folosească **GWYhostname1** sau **GWYhostname2**. O dată ce decizia este luată, clientul are o conexiune socket directă la una din aceste două porturi DB2 Connect. O dată ce conexiunea socket este realizată la serverul ales DB2 Connect, aveți un client tipic de server DB2 Connect la conectivitatea DB2 z/OS.

De exemplu, presupunând că distribuitorul alege **GWYhostname2**. Aceasta produce următorul mediu:

Client → DB2 Connect Server 2 → DB2 z/OS

Distribuitorul nu reîncearcă nici o conexiune dacă există orice tip de eșec de comunicații. Dacă doriți să activați caracteristica de rerutare client automată pentru o bază de date într-un asemenea mediu, serverul alternativ pentru baza de date asociată sau bazele de date din serverul DB2 Connect (DB2 Connect Server 1 sau DB2 Connect Server 2) ar trebui setat să fie distribuitor. (**DThostname**). Apoi, dacă DB2 Connect Server 1 se blochează din orice motiv, rerutarea de client automată reacționează și o conexiune client este reîncercată cu distribuitorul atât pe serverul primar cât și pe serverul alternativ. Această opțiune vă permite să combinați și să mențineți capacitățile de distribuitor cu caracteristica de rerutare de client automată DB2. Setarea serverului alternativ la o gazdă diferită de numele de gazdă de distribuitor asigură clienții cu caracteristica de rerutare de client automată. Totuși, clienții vor stabili conexiuni directe la serverul alternativ definit și vor ocoli tehnologia de distribuitor, ceea ce elimină distribuitorul și valoarea pe care el o aduce.

Caracteristica de rerutare client automată interceptează următoarele coduri SQL:

- sqlcode -20157
- sqlcode -1768 (cod motiv = 7)

Notă: Rerutarea de client poate să nu fie informată de eșecurile de socket la acest moment de timp dacă setarea parametrului de configurare sistem de operare "TCP Keepalive" este prea înaltă. (Notați că numele acestui parametru de configurare diferă cu platforma.)

Partea 4. Ajustarea și DB2 Connect

Există considerente speciale în ceea ce privește performanța unui mediu de lucru în care se folosește DB2 Connect pentru transferul cererilor și răspunsurilor între stațiile de lucru client și serverele de bază de date. Îmbunătățirea sau menținerea performanței în acest mediu se face în mai multe feluri.

Capitolul 14. Considerente privind performanța DB2 Connect

Performanța este modul în care se comportă un sistem în cazul unei anumite încărcări de lucru. Este afectată de resursele disponibile și de modul în care acestea sunt utilizate și partajate. Dacă doriți să îmbunătățiți performanța, trebuie mai întâi să decideți ce fel de performanță doriți.

Puteți alege mai multe *metrice de performanță* diferite, cum ar fi:

Timpul de răspuns

Intervalul de timp dintre momentul în care aplicația trimite bazei de date cererea și momentul în care aplicația primește un răspuns.

Debitul tranzacției

Numărul unităților de lucru care pot fi completate într-o unitate de timp. Unitatea de lucru poate fi simplă, cum ar fi citirea și actualizarea unui rând, sau complicată, implicând sute de instrucțiuni SQL.

Rata transferului de date

Numărul de octeți de date transferați între aplicația DB2 Connect și baza de date mainframe IBM pe unitatea de timp.

Performanța va fi limitată de resursele hardware și software disponibile. Printre exemplele de resurse hardware se numără CPU-ul, memoria și adaptoarele de rețea. Printre resursele software se numără subsistemele de comunicații, subsistemele de paginare, mbuf pentru AIX.

Fluxurile de date

Figura 12 la pagina 86 arată calea fluxului de date care circulă între serverul de bază de date mainframe IBM și stația de lucru prin DB2 Connect.

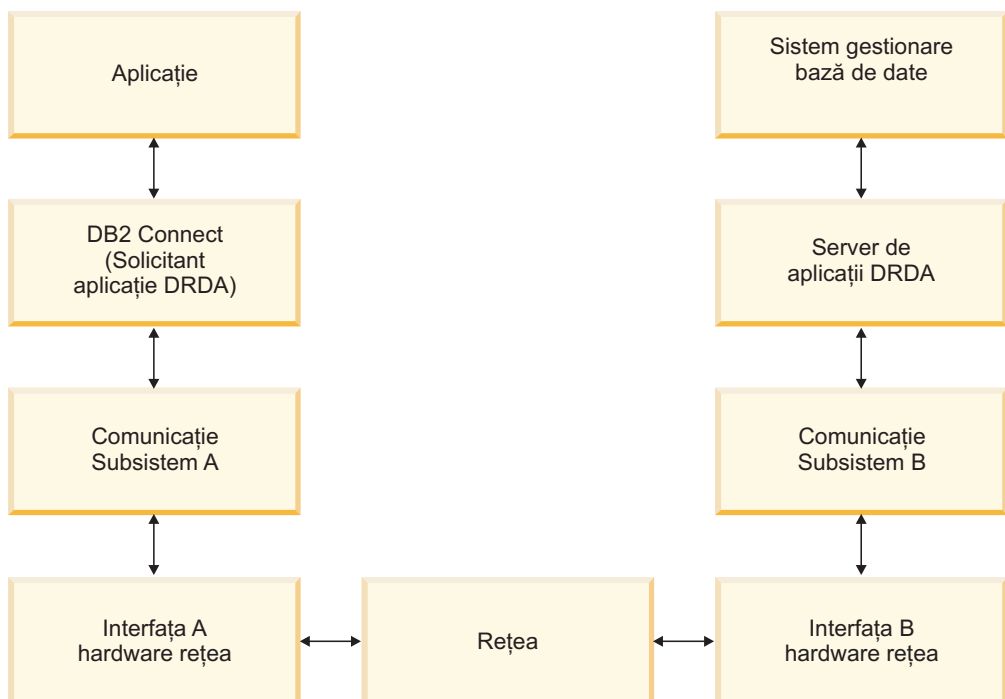


Figura 12. Fluxurile de date în DB2 Connect

- Baza de date mainframe IBM și o parte din subsistemul de comunicație B rulează de obicei pe același sistem. Acest sistem este alcătuit dintr-unul sau mai multe CPU, memorie principală, un subsistem I/O, DASD și un sistem de operare. Deoarece alte programe ar putea partaja aceste componente, conflictele pentru resurse ar putea cauza probleme de performanță.
- Rețeaua este compusă dintr-o combinație de cabluri, hub-uri, linii de comunicație, switch-uri și alte controlere de comunicație. De exemplu, interfața hardware de rețea B ar putea fi controlere de comunicație ca 3745 sau 3172 sau un adaptor token ring pentru un server IBM Power Systems. Ar putea fi implicate mai multe medii de transmisie între interfețele hardware de rețea A și B.
- Interfața hardware de rețea A ar putea fi Token-Ring, Ethernet**, alt adaptor LAN sau un adaptor care suportă protocoale SDLC sau X.25.
- DB2 Connect și subsistemul de comunicații A se află de obicei pe același sistem. Pentru această discuție, s-a presupus că și aplicația se află pe același sistem.

Gâtuirile

Debitul tranzacției este dependent de componenta cea mai lentă din sistem. Dacă identificați o gâtuire a performanței, puteți ocoli adesea problema prin modificarea parametrilor de configurare, alocând mai multe resurse componentei care cauzează probleme, actualizând componenta sau adăugând o componentă nouă, pentru a reduce încărcarea de lucru.

Puteți utiliza diverse unelte pentru a determina cât timp consumă o interogare la fiecare componentă. Aceasta vă va da o idee despre ce componente ar trebui ajustate sau modernizate pentru a îmbunătăți performanța. De exemplu, dacă determinați că o interogare consumă 60% din timpul său pe mașina DB2 Connect, ați putea dori să reglați DB2 Connect sau (dacă aveți clienți la distanță) să adăugați o altă mașină DB2 Connect rețelei dumneavoastră.

Benchmark-ul

Benchmark-ul compară performanțele dintr-un mediu cu performanțele din altul.

Benchmark-ul poate începe prin rularea aplicației de test într-un mediu normal. Pe măsură ce problema performanței este izolată, pot fi dezvoltate cazuri specializate de test, pentru a limita domeniul funcției care este testată și observată.

Benchmark-ul nu trebuie să fie complex. Cazurile de test specializate nu trebuie să emuleze o aplicație întreagă pentru a obține informații care pot fi valorificate. Porniți cu măsurători simple și creșteți complexitatea doar când este justificată.

Caracteristicile unui benchmark bun:

- Fiecare test poate fi repetat.
- Fiecare iterație a unui test este pornită cu aceeași stare a sistemului.
- Hardware-ul și software-ul utilizat pentru benchmarking se potrivește cu mediul produsului.
- Nu există funcții sau aplicații active în sistem în afară de cele măsurate, dacă scenariul nu include unele activități care rulează în sistem.

Notă: Aplicațiile care sunt pornite utilizează memoria chiar și atunci când sunt minimizate sau neutilizate. Aceasta ar putea cauza paginarea și decalarea rezultatelor benchmark-ului.

Unelte de performanță

Următoarele tabele listează unele din uneltele care vă pot ajuta să măsurați performanța sistemului. Deoarece aceste unelte utilizează resurse se sistem, s-ar putea să nu le doriți active tot timpul.

Tabela 14. Unelte de performanță pentru CPU și utilizarea memoriei

Sistem	Unealtă	Descriere
AIX	vmstat, time, ps, tprof	Furnizează informații despre CPU sau problemele memoriei pe stația de lucru DB2 Connect sau clienți la distanță.
HP-UX	vmstat, time, ps, monitor și glance dacă este disponibil	
Windows	Microsoft Performance Monitor	

Tabela 15. Unelte de performanță pentru activitatea bazei de date

Sistem	Unealtă	Descriere
Toate	Monitorizare baza de date	Determină dacă problema vine de la baza de date.
System z	IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS, ASG-TMON for DB2 (ASG) și CA Insight Performance Monitor for DB2 for z/OS (Computer Associates International, Inc.)	
Windows	Microsoft Performance Monitor	

Tabela 16. Unele de performanță pentru activitatea rețelei

Sistem	Unealtă	Descriere
AIX	netpmon	Raportează statistici rețea de nivel scăzut, inclusiv statistici TCP/IP cum ar fi numărul de pachet sau cadrele primite pe secundă.
Controler de rețea ca și 3745	NetView Performance Monitor	Raportează utilizarea controlului de comunicare și VTAM.
Linux și UNIX	netstat	Manipularea traficului TCP/IP.

Capitolul 15. Optimizarea accesului ODBC

Baza de date DB2 furnizează optimizare specială proiectată pentru a îmbunătăți performanța comunicației prin ODBC. Aceste îmbunătățiri sunt disponibile pentru Microsoft Access, Lotus Approach sau Visual Basic. Puteți câștiga un transfer mai rapid ODBC utilizând DB2 Configuration Assistant (CA).

Pentru a activa ODBC optimizat:

- Dacă definiți o conexiune nouă:
 1. Porniți DB2 CA.
 2. Deschideți meniul Selected și selectați Add Database Using Wizard...
 3. Urmați paginile vrăjitorului până ajungeți la pagina **Data Source**.
 4. Bifați **Register this database for CLI/ODBC**.
 5. Specificați cum ar trebui înregistrate aplicațiile CLI/ODBC care accesează această bază de date:
 - **As system data source** înseamnă că baza de date este disponibilă tuturor utilizatorilor din sistem.
 - **As user data source** înseamnă că sunteți singurul utilizator care poate accesa baza de date.
 - **As file data source** înseamnă că va fi creat un fișier care conține informațiile sursă de date. Acest fișier sursă de date poate fi partajat cu alte stații de lucru dacă aveți o conexiune TCP/IP. Altfel, fișierul poate fi doar folosit pe acest calculator
 6. Tastați un **Nume sursă date**.
 7. (Opțional) Selectați o aplicație din lista **Optimize for application** pentru a optimiza setările de sursă date pentru o anumită aplicație.
 8. Faceți clic pe **OK** și ieșiți din CA.
- Dacă actualizați o conexiune existentă:
 1. Porniți DB2 CA.
 2. Faceți dublu clic pe aliasul de bază de date pe care doriți să-l optimizați.
 3. Faceți clic pe **Data Source**.
 4. Bifați **Register this database for CLI/ODBC**.
 5. Specificați cum ar trebui înregistrate aplicațiile CLI/ODBC care accesează această bază de date:
 - **As system data source** înseamnă că baza de date este disponibilă tuturor utilizatorilor din sistem.
 - **As user data source** înseamnă că sunteți singurul utilizator care poate accesa baza de date.
 - **As file data source** înseamnă că va fi creat un fișier care conține informațiile sursă de date. Acest fișier sursă de date poate fi partajat cu alte stații de lucru dacă aveți o conexiune TCP/IP. Altfel, fișierul poate fi doar folosit pe acest calculator
 6. Tastați un **Nume sursă date**.
 7. (Opțional) Selectați o aplicație din lista **Optimize for application** pentru a optimiza setările de sursă date pentru o anumită aplicație.
 8. Faceți clic pe **OK** și ieșiți din CA.

Capitolul 16. Proiectarea aplicației

Când creați o aplicație, puteți îmbunătăți performanțele în mai multe moduri.

Proceduri SQL compuse și memorate

Pentru aplicațiile care trimit și primesc multe comenzi și răspunsuri, regia de rețea poate fi semnificativă. Procedurile SQL compuse și memorate sunt două moduri de a reduce această regie.

Dacă o aplicație trimite mai multe instrucțiuni SQL fără intervenția logicii de program, puteți utiliza proceduri SQL compuse. Dacă aveți nevoie de logica de program în grupul de instrucțiuni SQL, puteți utiliza procedurile memorate.

Toate instrucțiunile executabile pot fi conținute în instrucțiunile compuse SQL, cu excepția următoarelor:

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

Procedurile memorate vă ajută la reducerea traficului în rețea prin plasarea programului logic pe server. Puteți să realizați comitere automat când ieșiți din procedură. Puteți de asemenea să întoarceți seturi de rezultate, care minimizează logica aplicației la client.

Cerințe pentru grupare

Gruparea cererilor de bază de date (instrucțiuni SQL) înrudite poate reduce numărul de cereri și răspunsuri transmise prin rețea.

De exemplu, grupând următoarele instrucțiuni:

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

în

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

sunt trimise mai puține cereri de-a lungul rețelei.

Puteți folosi de asemenea cuvinte cheie cum ar fi `IN` și `BETWEEN` pentru a reduce numărul de rânduri returnate. În plus, puteți folosi cuvintele cheie `WHERE`, `IN` și `BETWEEN` în instrucțiunile `UPDATE` și `DELETE`.

Predicatul logic

Puteți utiliza predicatul logic pentru a cere doar liniile și coloanele care sunt necesare. Aceasta minimizează traficul în rețea și regia CPU pentru transmisia de date.

De exemplu, nu utilizați interogarea:

```
SELECT * FROM TABLEA
```

dacă este nevoie numai de primul rând din TABLEA cu ROW_ID=1 sau numai de coloana 1 și coloana 2.

Gruparea datelor în blocuri

Ar trebui să utilizați gruparea datelor în blocuri dacă așteptați cantități mari de date de la server. Gruparea în blocuri îmbunătățește folosirea lățimii de bandă a rețelei și reduce regia CPU pe serverul de bază de date mainframe IBM și serverul DB2 Connect. Există o cantitate fixată a regiei CPU și a rețelei pentru fiecare mesaj trimis și recepționat indiferent de mărime. Blocarea datelor reduce numărul mesajelor necesare pentru transferul aceleiași cantități de date.

La blocare, prima linie de date dintr-un chestionar nu va fi livrat aplicației până este recepționat primul bloc. Blocarea crește timpul de răspuns pentru prima linie, dar îmbunătățește timpul de răspuns pentru liniile din subsecvență.

Un alt considerent este cantitatea de memorie utilizată. De obicei setul de memorie în lucru crește în momentul în care este setată blocarea.

Cu DB2 Connect, puteți controla cantitatea de date care este transferată la fiecare bloc.

Pentru a invoca gruparea pe blocuri, folosiți opțiunea BLOCKING a comenzii prep sau bind. Blocarea este setată, dacă:

- Cursorul este numai-citire sau
- Cursorul este ambiguu și este specificată blocarea în prep sau bind.

Notă: La utilizarea SQL dinamic, cursorul este totdeauna ambiguu.

Instrucțiunile SQL cu BLOCKING

Instrucțiunile SELECT actualizabile (folosind instrucțiuni UPDATE/DELETE WHERE CURRENT OF) sunt interogări neblocante, deci ar trebui să le folosiți doar când este absolut necesar.

O instrucțiune SELECT actualizabilă împiedică modificarea rândului între momentul finalizării instrucțiunii SELECT și cel al lansării UPDATE/DELETE. Dacă acest nivel de concurență nu este important pentru aplicația dumneavoastră, o alternativă este utilizarea a DELETE sau UPDATE împreună cu criteriile de căutare bazate pe valorile returnate de către un SELECT care nu poate fi actualizat.

Pentru SELECT numai-citire, specificați FOR FETCH ONLY, dar nu pe VM și VSE, unde nu este suportat.

SQL static și dinamic

Utilizați SQL static cât mai mult posibil. Evită pregătirea secțiunii SQL run-time și cursoarele ambigue. Dacă nu poate fi evitat SQL dinamic, puteți face următoarele pentru a minimiza traficul în rețea și pentru a îmbunătăți performanțele:

- Dacă instrucțiunea este SELECT și trebuie să fie pregătită, realizați PREPARE ... INTO SQLDA. SQLDA trebuie alocată la dimensiunea întreagă necesară setărilor dumneavoastră. Dacă numărul maxim al coloanelor este x și se așteaptă să rămână așa, alocați o SQLDA cu x SQLVAR-uri. Dacă numărul coloanelor potențiale este nesigur (și nu este nici o problemă în ce privește memoria), utilizați numărul maxim de SQLVAR-uri (256).

Dacă alocarea SQLDA nu este destul de mare pentru a memora ceea ce întoarce SQLDA, programul trebuie să apeleze un alt DESCRIBE cu o SQLDA destul de mare pentru a memora rezultatul din nou. Aceasta va crește traficul din rețea.

Nu utilizați secvențele PREPARE și DESCRIBE. Utilizarea instrucțiunilor PREPARE.....INTO asigură o performanță mai bună.

- Executați legături statice ale instrucțiunilor SQL COMMIT sau ROLLBACK în locul instrucțiunilor dinamice COMMIT sau ROLLBACK.
- Dacă nu este o instrucțiune SELECT, COMMIT sau ROLLBACK, pentru a executa instrucțiunea lansați EXECUTE IMMEDIATE, în locul secvenței PREPARE și EXECUTE.
- Aplicațiile ODBC utilizează SQL dinamic. Puteți utiliza caracteristica de profilare statică CLI/ODBC pentru îmbunătățirea performanței. Această caracteristică vă permite să capturați și să convertiți apelurile ODBC la instrucțiuni statice memorate într-un pachet al bazei de date. Performanțele reale pe care le veți obține depind de complexitatea aplicației dumneavoastră.

Alte considerente SQL

Utilizând Command Line Processor (CLP) este, în general, mai încet decât având SQL dinamic în program deoarece CLP trebuie să analizeze sintactic intrarea înaintea lansării în execuție a motorului bazei de date SQL. CLP formatează de asemenea datele când sunt primite, ceea ce poate să nu fie necesar pentru aplicația dumneavoastră.

Instrucțiunile SQL într-un limbaj interpretat, cum ar fi REXX, sunt substanțial mai încete decât aceleași instrucțiuni SQL într-un limbaj compilat, cum ar fi C.

Există două tipuri de instrucțiuni CONNECT numite tipul 1 și tipul 2. La conectarea de tip 2, conectarea la o bază de date pune conexiunea anterioară într-o stare de adormire dar nu o abandonează. Dacă veți comuta ulterior la o conectare adormită, evitați regia de încărcare a bibliotecilor și setarea structurilor de date interne. Din acest motiv, folosirea conectării de tip 2 ar putea îmbunătăți performanța pentru aplicații care accesează mai mult de o bază de date.

Capitolul 17. Gestionarea conexiunilor

Gestionarea conexiunilor constă din două operații: punerea conexiunilor în pool și concentratorul de conexiuni. Punerea conexiunilor în pool reduce regia pentru conexiunile bazei de date și gestionează volumul de conexiuni. Concentratorul de conexiuni crește scalabilitatea mediului dumneavoastră de lucru prin optimizarea resurselor folosite de serverele de bază de date gazdă. Aici sunt discutate ambele operații.

Punerea în pool a conexiunilor

Produsele server DB2 Connect, cum ar fi DB2 Connect Enterprise Edition, furnizează adesea conexiuni la baza de date pentru mii de cereri clienți simultan. Stabilirea și servirea conexiunilor la serverul de bază de date poate fi un proces care cere multe resurse care afectează grav performanța serverului bază de date și serverului DB2 Connect.

Această problemă este evidentă mai ales în mediile web, unde fiecare vizită într-o pagină web poate necesita construirea unei noi conexiuni la serverul bazei de date, efectuarea unei interogări și terminarea conexiunii. Pentru a reduce această regie, produsele server DB2 Connect folosesc punerea conexiunilor în pool, pentru a menține conexiunile deschise la baza de date într-un pool gata accesibil.

Majoritatea aplicațiilor bazate pe tehnologii web execută un volum larg de tranzații scurte. O tranzație web tipică este executată ca parte a propriei conexiuni. Cu alte cuvinte, executarea unei tranzații înseamnă stabilirea unei conexiuni la baza de date și apoi terminarea aceste conexiuni după doar câteva instrucțiuni SQL. Acest proces de stabilire și distrugere a conexiunii este foarte costisitor. Implică crearea unui agent DB2 Connect, stabilirea unei conexiuni în rețea între acest agent și serverul DB2 și crearea unui fir de execuție DB2 pe server. Pentru conexiuni cu rulare mai lungă aceste costuri sunt amortizate peste toate tranzațiile executate în această conexiune, dar pentru o tranzație web tipică, aceste costuri vor depăși costul execuției tranzației propriu-zise.

Punerea în pool a conexiunilor este o tehnică ce permite reutilizarea unei infrastructuri de conexiune care a fost stabilită, pentru conexiuni ulterioare. Când este pornită o instanță DB2 Connect, este creat un pool de agenți coordonatori. Când vine o cerere de conexiune este alocat un agent acestei cereri. Acest agent se va conecta la serverul DB2 și va fi creat un fir de execuție în DB2. Când aplicația emite o cerere de deconectare, agentul nu va transmite această cerere mai departe la serverul DB2. În loc, agentul este pus înapoi în pool. Agentul din pool este în continuare proprietarul conexiunii sale la serverul DB2 și al firului de execuție DB2 corespunzător. Dacă o altă aplicație lansează o cerere de conectare, acest agent este atribuit noii aplicații. Pentru ca operarea să fie sigură, informațiile privind identitatea utilizatorului sunt transmise mai departe la firul de execuție DB2 care, la rândul lui, realizează autentificarea utilizatorului.

Pool-ul de conexiuni DB2 asigură o îmbunătățire semnificativă a performanței în astfel de medii. DB2 Connect menține conexiunile deschise la baza de date într-un pool disponibil. Când un client cere o conexiune, ea poate fi oferită din acest pool de conexiuni pregătite. Punerea conexiunilor în pool reduce semnificativ regia (activitatea suplimentară) pe care o implică de obicei deschiderea și închiderea acestor conexiuni.

Punerea conexiunilor în pool este transparentă pentru aplicațiile care se conectează la gazdă prin DB2 Connect. Când o aplicație cere deconectarea de la gazdă, DB2 Connect abandonează conexiunea de intrare cu aplicația, dar păstrează conexiunea de ieșire la gazdă

într-un pool. Când o aplicație nouă cere o conexiune, DB2 Connect folosește una din pool-ul existent. Folosind o conexiune deja prezentă, se reduce timpul de conexiune general, precum și costul înalt de conectare CPU de pe gazdă.

Agenții DB2 Connect pot fi în două stări: nefolosiți sau activi. Un agent este activ când lucrează pentru o aplicație. După terminarea lucrului, agentul intră în starea nefolosit, așteptând altă sarcină de la aceeași aplicație sau de la alta. Toți agenții nefolosiți sunt păstrați la un loc, în ceea ce este cunoscut ca pool-ul de agenți nefolosiți. Puteți configura dimensiunea acestui pool folosind parametrul de configurare **num_poolagents**. Acest parametru este egal cu numărul maxim de agenți nefolosiți pe care vreți să îi mențină sistemul dumneavoastră. Setarea acestui parametru la 0 este echivalentă cu dezactivarea opțiunii de punere în pool a conexiunilor. Valoarea implicită pentru acest parametru este setată la AUTOMATIC cu o valoare de 100. Fiind setată la AUTOMATIC, DB2 Connect gestionează automat numărul de agenți nefolosiți (idle) în pool-ul de agenți nefolosiți.

DB2 Connect nu stabilește conexiuni la baza de date înainte de a primi prima cerere client. Alternativ, puteți umple pool-ul de agenți nefolosiți înainte care vreun client să facă o cerere. Pool-ul poate fi completat la pornire folosind parametrul de configurare **num_initagents**. Acest parametru determină câți agenți nefolosiți trebuie creați la pornire. Acești agenți nefolosiți inițial vor avea conexiuni la serverul de bază de date gazdă.

Când un client cere o conexiune la gazdă, DB2 Connect va încerca să obțină un agent din cei din pool care au o conexiune la serverul de bază de date gazdă. Dacă aceasta eșuează, va încerca să găsească un agent disponibil în pool-ul cu agenți nefolosiți. Dacă pool-ul este gol, DB2 Connect va crea un agent nou.

Puteți controla numărul maxim de agenți care sunt activi concurent, folosind parametrul de configurare **max_coordagents**. După ce acest număr este depășit, noile conexiuni vor eșua cu sqlcode SQL1226. (Acest cod înseamnă că numărul maxim de conexiuni de ieșire a fost depășit.) Valoarea implicită pentru acest parametru este setată la AUTOMATIC cu o valoare de 200. Fiind setată la AUTOMATIC, DB2 Connect gestionează automat numărul de agenți coordonatori.

Variabila de registru DB2 DB2CONNECT_IN_APP_PROCESS permite aplicațiilor care rulează pe aceeași mașină ca produsul server DB2 Connect să aibă ori DB2 Connect să ruleze în procesul aplicațiilor, comportarea implicită sau să determine ca aplicația să se conecteze la produsul server DB2 Connect și conexiunea gazdă să ruleze într-un agent. Pentru ca o aplicație să folosească punerea în pool a conexiunilor, conexiunile la gazdă trebuie făcute din agenții produsului server DB2 Connect și deci DB2CONNECT_IN_APP_PROCESS trebuie setat la NO.

Punerea în pool a conexiunilor DB2 Connect față de punerea în pool a conexiunilor serverului de aplicații

Punerea în pool a conexiunilor este o necesitate pentru orice tehnologie web bazată pe aplicații care suportă un volum mare de tranzacții. Majoritatea serverelor de aplicații web oferă acum propriul mod de a pune în pool conexiunile la baza de date. De exemplu, atât Microsoft MTS (COM+), cât și IBM WebSphere asigură punerea conexiunilor în pool.

Mecanismele de punere în pool implementate de aceste servere diferă semnificativ de ceea ce oferă serverele DB2 Connect. Deoarece serverele de aplicații pun conexiunile în pool numai pentru uz propriu, ele presupun că ID-ul de utilizator, parola, nivelurile de izolare și așa mai departe vor fi exact la fel pentru toate conexiunile. Și mai important este faptul că serverele de aplicații pun în pool numai conexiunile inițiate de același proces. Aceasta înseamnă că nu sunt puse în pool conexiunile de la alte mașini, utilizatori sau procese. În timp ce aceste tehnici ale serverului de aplicații de punere în pool sunt eficiente pentru re folosirea conexiunilor stabilite

de aceeași instanță a unei aplicații, ele sunt absolut ineficiente pentru punerea în pool a conexiunilor de la mai mulți utilizatori, servere și așa mai departe.

Punerea în pool a conexiunilor pe care o asigură serverele DB2 Connect este complet independentă de aplicație, mașină și utilizator. Conexiunile de la mai mulți clienți, servere de aplicații, toate cu ID-uri de utilizator diferite, pot de asemenea refolosi conexiunile altora, fapt care are ca rezultat utilizarea mult mai bună a resurselor puse în pool.

Ce tip de punere în pool a conexiunilor este cea corectă pentru utilizare? Ambele. În general, folosirea atât a punerii în pool a conexiunilor DB2 Connect, cât și a punerii în pool a conexiunilor serverului de aplicații este o strategie bună, deoarece nu interferează între ele. Chiar când este activată punerea în pool a conexiunilor serverului de aplicații, punerea în pool a conexiunilor DB2 Connect poate asigura refolosirea conexiunii pentru mai multe servere de aplicații, precum și pentru alți clienți care folosesc serverul DB2 Connect.

Concentratorul de conexiuni

Concentratorul de conexiune reduce resursele necesare pe serverele de bază de date DB2 for z/OS pentru a suporta numere mari de stații de lucru și utilizatori web. Această funcție poate crește dramatic scalabilitatea DB2 for z/OS și soluției DB2 Connect furnizând de asemenea balansarea încărcării de lucru la nivel tranzacție și operații sigure în DB2 pentru medii de partajare de date DB2 for z/OS.

Concentratorul de conexiuni permite aplicațiilor să stea conectate fără a fi consumate resurse pe serverul gazdă DB2. Puteți avea mii de utilizatori activi în aplicații și să aveți doar câteva fire de execuție active pe serverul gazdă DB2.

Tehnologia *concentratorului de conexiuni* a lui DB2 Connect permite produselor server DB2 Connect, cum ar fi DB2 Connect Enterprise Edition, să furnizeze suport pentru mii de utilizatori care să execute simultan tranzacții de afaceri, reducând drastic resursele necesare pe serverele de bază de date gazdă System z sau IBM Power Systems. Realizează acest scop concentrând sarcina de lucru de la toate aplicațiile într-un număr mult mai mic de conexiuni S/390 server bază de date gazdă System z sau IBM Power Systems. Chiar dacă aceasta ar putea avea similitudini cu funcția de punere a conexiunilor în pool descrisă mai sus, este de fapt o abordare mai sofisticată a reducerii consumului de resurse pentru aplicații cu volum mare de OLTP.

Concentratorul de conexiuni preia conceptul unui agent și îl împarte în două entități:

- Agentul logic, care reprezintă o conexiune de aplicație.
- Agentul coordonator, care posedă conexiunea și firul de execuție DB2 și execută cereri aplicație.

Când o nouă aplicație încearcă o conectare la gazdă, îi este alocat un agent logic. Pentru a transmite cod SQL la baza de date este necesar un agent coordonator și acesta este alocat de îndată ce este inițiată o nouă tranzacție. Cheia acestei arhitecturi este faptul că agentul coordonator este:

- Disociat de agentul logic
- Returnat în pool când tranzacția se încheie ca urmare a unei comiteri sau derulări înapoi

Altă caracteristică cheie este metoda de alocare a agenților coordonatori la noi tranzacții într-un mediu cu partajarea datelor. DB2 Connect implementează un algoritm de planificare sofisticat care folosește informații System z Work Load Manager (WLM). Aceste informații sunt folosite pentru a distribui sarcina de lucru de-a lungul membrilor unui grup de partajare date în funcție de criteriile setate în WLM. WLM nu numai că știe încărcătura fiecărui membru, dar le știe și disponibilitatea. Aceasta permite DB2 Connect să realoce transparent lucrări de la membrii eșuați sau supraîncărcați la membrii care există și sunt subutilizați.

Concentratorul de conexiuni DB2 Connect este activat când setați numărul maxim de agenți logici (*max_connections*) mai mare decât numărul de agenți de coordonare (*max_coordagents*).

Pool-ul de conexiuni reduce costul stabilirii conexiunii în caz că una nu mai este necesară prin terminarea unei aplicații. Cu alte cuvinte, o aplicație trebuie să se deconecteze înainte ca altcineva să folosească din nou conexiunea pusă în pool.

Alternativ, concentratorul de conexiuni permite DB2 Connect să facă o conexiune disponibilă unei aplicații de îndată ce altă aplicație a terminat o tranzacție și nu necesită deconectarea altei aplicații. În esență, o conexiune la serverul de bază de date și resursele sale gazdă și DB2 Connect asociate sunt folosite de o aplicație doar când are o tranzacție activă. De îndată ce tranzacția s-a încheiat, conexiunea și resursele asociate sunt disponibile pentru a fi folosite de orice altă aplicație în care trebuie să fie executată o tranzacție.

În versiunile anterioare ale DB2 Connect, fiecare aplicație activă avea un Engine Dispatchable Unit (EDU) care gestiona conexiunea la baza de date precum și orice cereri de aplicație. Acest EDU era numit de obicei *agent coordonator*. Fiecare agent coordonator urmărea starea sau contextul aplicației și al EDU-ului. Fiecare EDU ocupă o cantitate semnificativă de memorie când crește numărul de conexiuni, iar comutarea de context între agenți are ca rezultat o regie suplimentară.

În arhitectura de mai sus, există o relație unu la unu între conexiuni și EDU. Concentratorul de conexiuni permite însă relații multe-la-unu între conexiuni și EDU-uri. Cu alte cuvinte, relația dintre conexiuni (X) și EDU-uri (Y) este acum $X \geq Y$.

Concentratorul de conexiuni împarte agentul în două entități, un *agent logic* și un *agent lucrător*. Agenții logici reprezintă o aplicație, dar fără referință la un EDU particular. Agentul logic conține toate informațiile și blocurile de control necesare aplicației. Dacă există *n* aplicații conectate la server, vor fi *n* agenți logici pe server. Agenții lucrători sunt EDU fizici care execută cereri de aplicații, dar care nu au atașament permanent la nici o aplicație dată. Agenții lucrători se asociază cu agenți logici pentru a realiza tranzacții și la limita tranzacției termină asocierea și se întorc la pool-ul disponibil.

O entitate cunoscută ca *dispecer* alocă agenții lucrători la agenții logici. Limitările numărului de mâneri de fișiere deschise pe anumite platforme de calcul ar putea avea ca rezultat mai mult de o instanță de planificator.

Restricțiile pentru concentratorul de conexiuni

Există un număr de restricții importante pentru folosirea concentratorului serverului DB2 Connect. Citiți următoarele informații în totalitate înainte de a încerca să utilizați concentratorul de conexiuni în sistem.

Restricții generale:

- Concentratorul se bazează pe protocolul TCP/IP pentru a stabili conexiuni de intrare de la clienții locali sau la distanță. Doar conexiunile de intrare care folosesc TCP/IP sau Local (IPC) vor putea folosi conexiuni de ieșire puse în pool. Concentratorul va accepta conexiuni prin alte protocoale de comunicație, cum ar fi Named Pipes, dar nu veți putea să îi folosiți caracteristicile de concentrare XA cu o astfel de conexiune.
- Pentru suport de tranzacție XA strâns cuplată, toate aplicațiile care participă în aceeași tranzacție XA trebuie să folosească aceeași instanță de server DB2 Connect pentru a se conecta la gazdă.
- Doar aplicațiile care închid resursele deținute (cum ar cursoarele deținute) la granița de tranzacție pot beneficia de concentrator. Tranzacțiile care nu închid cursoarele deținute vor

merge în continuare, dar vor fi alocate unui agent lucrător dedicat și astfel nu vor putea utiliza întregul set de caracteristici al concentratorului.

- Dacă declarați tabele temporare, acestea trebuie abandonate explicit la limita tranzacției sau ramificării. Dacă nu se abandonează tabelele, se va opri concentrarea conexiunilor, dar aplicația va continua să lucreze.
- Toate aplicațiile care participă la aceeași tranzacție XA trebuie să aibă același CCSID și să folosească același ID de utilizator pentru a efectua conexiunea.
- Dacă este stabilită o conexiune de ieșire pentru a suporta o conexiune în două faze, agentul acelei conexiuni poate fi folosit doar pentru a suporta conexiuni în două-faze. Similar, agenții stabiliți să suporte o conexiune cu o singură fază pot suporta doar conexiune o-fază.
- Concentratorul suportă aplicațiile care folosesc IBM Data Server Driver for JDBC and SQLJ și aplicațiile CLI (Call Level Interface) care folosesc SQL dinamic. De asemenea, aplicațiile CLI nu trebuie să folosească KEEP DYNAMIC, deoarece concentratorul depinde de instrucțiunile care sunt pregătite din nou în granițele fiecărei tranzacții.
- Cererile de pregătire dinamice din aplicații SQL dinamic încorporat vor fi respinse. Aplicațiile trebuie să fie modificate astfel încât să utilizeze QSL static sau CLI pentru instrucțiuni SQL dinamic.
- Dacă concentratorul de conexiuni este ON, cererea de intrare la serverul DB2 Connect nu poate folosi SSL. Totuși, cererea de ieșire la serverul de bază de date destinație poate folosi SSL. Dacă concentratorul de conexiuni este OFF, amândouă tipurile de cereri, de intrare și de ieșire, pot folosi SSL.

Când se lucrează cu DB2 Versiunea 9 sau Versiunea 8 FixPak 13 (sau mai mare), pentru a activa suportul concentratorului DB2 Connect este nevoie de IBM Power Systems Versiunea 5 Ediția 4 (PTF SI23726). În caz contrar, este suportată numai porțiunea XA a concentratorului de conexiuni.

Activarea concentratorului de conexiuni

Parametrul de configurare a managerului bazei de date *max_coordagents* setează numărul maxim de agenți logici. Puteți activa opțiunea de concentrator prin setarea valorii *max_connections* la orice număr mai mare decât cel implicit. Valoarea implicită pentru *max_connections* este echivalentă cu valoarea *max_coordagents*. Deoarece fiecare aplicație va avea un agent logic, *max_connections* controlează numărul de aplicații care pot fi conectate la instanța bazei de date, în timp ce *max_coordagents* controlează numărul de conexiuni de intrare care pot fi active în orice moment. *max_connections* va lua o valoare numerică în intervalul de la *max_coordagents* la 64000. Numărul implicit de agenți logici este egal cu *max_coordagents*.

Se poate seta pe AUTOMATIC atât pentru *max_connections*, cât și pentru *max_coordagents*. Dacă *max_connections* se setează la AUTOMATIC, numărul de conexiuni poate fi crescut peste valoarea de bază configurată. Dacă s-a setat AUTOMATIC atât pentru *max_connections*, cât și pentru *max_coordagents*, *max_connections* poate crește peste valoarea de bază, iar *max_coordagents* crește automat pentru a menține raportul de concentrare dintre conexiuni și agenții coordonatori.

Pentru configurarea agenților sunt folosiți câțiva parametri de configurare existenți. Acești parametri sunt:

max_coordagents

Număr maxim de agenți de coordonare activi.

num_poolagents

Dimensiune pool de agenți. Pool-ul de agenți include agenți inactivi și neutilizați. Pentru o performanță mai bună, ar trebui să se configureze *num_poolagents* la numărul mediu de clienți.

num_initagents

Numărul inițial de agenți lucrători din pool. Aceștia vor fi agenți neutilizați.

suport de tranzacții XA

Arhitectura concentratorului de conexiuni permite DB2 Connect să furnizeze suport pentru tranzacții strâns cuplate XA pentru DB2 for z/OS și DB2 for IBM i. Concentratorul va asocia un agent lucrător cu o anumită tranzacție XA (XID unic) ca pentru orice altă tranzacție. Totuși, dacă tranzacția XA se încheie prin *xa_end()* (graniță ramificare), agentul lucrător nu se va autolansa în pool-ul general. În loc, lucrătorul rămâne asociat cu o tranzacție XA. Când o altă aplicație se unește cu aceeași tranzacție XA, agentul lucrător va fi atașat aplicației.

Orice apel la granița de tranzacție va returna agentul în pool. De exemplu, *xa_prepare()* cu acces numai citire, *xa_rollback()*, *xa_recover()*, *xa_forget()*, *xa_commit()* sau orice eroare XA care cauzează derularea înapoi va returna agentul în pool-ul normal. *Xa_end()* încheie numai ramificarea tranzacției, dar nu este suficient pentru a încheia asocierea cu XID.

Exemple de suport de tranzacții XA

1. Considerați un mediu în care sunt necesare 4000 de conexiuni concurente sau mai multe. Un server web care folosește aplicații CGI sau un sistem de birou cu mulți utilizatori desktop poate să depășească această cerință. În acele cazuri, eficiența va necesita ca DB2 Connect să opereze ca un gateway singular; adică, baza de date și sistemul DB2 Connect sunt pe mașini separate.

Sistemul server DB2 Connect se poate să nu poată susține 4000 de conexiuni deschise simultane la mașina bază de date. În majoritatea cazurilor, numărul de tranzacții care apar la un anumit moment va fi considerabil mai mic decât numărul de conexiuni concurente. Administratorul de sistem poate maximiza eficiența sistemului prin setarea parametrilor de configurație ai bazei de date după cum urmează:

```
MAX_CONNECTIONS = 4,000
MAX_COORDAGENTS = 1,000
NUM_POOLAGENTS  = 1,000
```

Concentratorul va păstra deschise până la 4000 de conexiuni concurente, deși gateway-ul poate gestiona numai 1000 de tranzacții la un moment dat.

2. În exemplul de mai sus, agenții lucrători vor forma și distruge în mod constant asociațiile la agenții logici. Agenții care nu sunt nefolosiți ar putea menține o conexiune la baza de date, dar nu participă la nici o tranzacție, de aceea sunt disponibili oricărui agent logic (aplicație) care cere o conexiune.

Cazul tranzacțiilor XA este diferit. Pentru acest exemplu, presupuneți că un monitor TP este folosit cu un gateway DB2 Connect și o bază de date System z sau IBM Power Systems. Când o aplicație cere o conexiune, concentratorul va întoarce un agent inactiv peste serverul care cere sau creează un nou agent lucrător. Să presupunem că aplicația cere o tranzacție XA. Este creat un XID pentru această tranzacție și agentul lucrător este asociat cu el.

Când cererea aplicației a fost servită, lansează *xa_end()* și se detașează de la agentul lucrător. Agentul lucrător rămâne asociat cu XID-ul tranzacției. Poate servi numai cereri pentru tranzacții cu XID-ul asociat.

În acest moment, altă aplicație ar putea face o cerere pentru o tranzacție non-XA. Chiar dacă nu sunt agenți lucrători disponibili, agentul asociat cu XID nu va fi făcut disponibil pentru a doua aplicație. Este considerat activ. Cea de-a doua aplicație va avea un nou

agent lucrător creat pentru aceasta. Când cea de-a doua aplicație își încheie tranzacția, agentul său lucrător este eliberat în pool-ul disponibil.

Între timp, alte aplicații care cer tranzacția asociată cu XID-ul primului agent se pot atașa și detașa de agent, care execută tranzacția dedicată XA pentru ele. Orice aplicație care cere ca această tranzacție particulară va fi trimisă către acest agent lucrător, dacă este liber.

Agentul lucrător nu va fi lansat înapoi în pool-ul general decât atunci când o aplicație emite un apel de graniță de tranzacție (nu `xa_end()`). De exemplu, o aplicație poate încheia tranzacția cu `xa_commit()`, în momentul în care agentul lucrător abandonează asocierea cu XID și se întoarce la pool-ul disponibil. La acest punct, orice cerere de aplicație poate fi folosită pentru o altă tranzacție XA sau non-XA.

Punerea conexiunilor în pool și concentratorul de conexiuni

Punerea conexiunilor în pool și concentratorul de conexiuni par să fie similare, dar diferă prin implementarea lor și prin problemele pe care le tratează. Punerea conexiunilor în pool ajută la reducerea regiei conexiunilor la baza de date și a volumului de conexiuni manipulate. Concentratorul de conexiuni ajută la creșterea scalabilității DB2 for z/OS și a soluției DB2 Connect optimizând folosirea serverelor dumneavoastră bază de date gazdă.

Când se utilizează punerea în pool a conexiunilor, conexiunea este disponibilă pentru reutilizare numai după ce aplicația care este proprietara conexiunii lansează o cerere de deconectare. În multe aplicații client-server pe 2 niveluri (2-tier) utilizatorii nu se deconectează pe durata unei zile de lucru. De asemenea, cele mai multe servere de aplicații din aplicațiile multinivel stabilesc conexiuni la baza de date atunci când pornește serverul și aceste conexiuni nu mai sunt eliberate decât la oprirea serverului de aplicații.

În aceste medii, punerea conexiunilor în pool va aduce puține avantaje, poate nici unul. Dar în mediile web și client-server în care frecvența conectărilor și deconectărilor este mare, punerea conexiunilor în pool va produce îmbunătățiri semnificative ale performanței. Concentratorul de conexiuni alocă resursele bazei de date gazdă doar pe durata unei tranzacții SQL, în timp ce aplicațiile de utilizator rămân active. Aceasta permite configurații în care numărul de fire de execuție DB2 și resursele pe care acestea le consumă să fie mult mai mici decât în cazul în care fiecare conexiune de aplicație are propriul său fir de execuție.

Când este vorba de operații sigure la eroare și despre echilibrarea încărcării de lucru, concentratorul de conexiuni este clar alegerea corectă, deoarece permite realocarea lucrului cu fiecare nouă tranzacție. Alternativ, punerea conexiunilor în pool poate oferi doar o echilibrare foarte limitată și doar la conectare.

Punerea conexiunilor în pool și concentratorul de conexiuni trebuie să fie utilizate împreună, deși se adresează unor probleme diferite.

Concentratorul de conexiuni este necesar cu WebSphere MQ Transaction Manager și DB2 for z/OS

Când sunt rulate aplicații într-un mediu de lucru IBM WebSphere MQ (numit anterior IBM MQSeries), WebSphere MQ poate acționa ca un manager de tranzacții compatibil XA, coordonând tranzacțiile distribuite cu comitere în două faze. Când WebSphere MQ acționează ca un manager de tranzacții în acest fel și sursele de date sunt din familia de produse DB2, există câteva cerințe de configurare.

Cele mai multe dintre cerințele de configurare dintr-un astfel de mediu de manager de tranzacții sunt deja documentate în altă parte. De exemplu, trebuie să setați parametrul de configurare DB2 `TP_MON_NAME` la "MQ" în clientul runtime DB2.

Lipsește însă o cerință de configurare. Această cerință este specifică pentru DB2 Connect când se realizează conectarea la sursele de date care sunt servere DB2 for z/OS: când se folosește WebSphere MQ pentru a coordona tranzacțiile distribuite ce implică servere DB2 for z/OS și DB2 for IBM i, pe gateway trebuie să fie activată caracteristica de concentrator de conexiuni DB2 Connect. Concentratorul de conexiuni este activat atunci când valoarea parametrului de configurare **max_connections** este mai mare decât valoarea parametrului de configurare **max_coordagents**.

Dacă nu activați concentratorul de conexiuni, poate rezulta un comportament neașteptat al tranzacțiilor.

Capitolul 18. Suportul Sysplex pentru serverul DB2 Connect

Un Sysplex este o colecție de servere System z care cooperează, folosind hardware și software, pentru a procesa lucrul. Sysplex coordonează cooperarea, crescând numărul de procesoare care lucrează împreună, ceea ce permite executarea mai multor lucrări. În afară de creșterea capabilității de procesare, un Sysplex poate oferi flexibilitate în ceea ce privește mixarea nivelurilor de hardware și de software și adăugarea dinamică a sistemelor.

Sysplex permite serverului DB2 Connect să echilibreze conexiunile între diferiții membri ai grupului de partajare a datelor. Sysplex îi furnizează serverului DB2 Connect mijloacele să încerce membri alternativ în cazul apariției unui defect la unul din membri. Capacitatea de rerutare Sysplex este o caracteristică a lui DB2 Connect. Suportul Sysplex pentru serverul DB2 Connect este activat implicit, ca și capabilitatea de rerutare pentru Sysplex. Suportul Sysplex pentru o bază de date gazdă poate fi dezactivat prin înlăturarea parametrului SYSPLEX din intrarea de director DCS, dar fără a înlătura intrarea DCS propriu-zisă, chiar dacă nu mai are alți parametri specificați.

Cu capabilitatea de rutare client automată pentru Sysplex, comportamentul implicit este ca o conexiune activată Sysplex să reîncerce conectarea când există un defect de comunicație. Valorile registrului special, până la ultima tranzacție cu succes care nu blochează resurse, sunt rulate din nou când DB2 Connect este conectat la un server DB2 for z/OS.

Puteți configura comportarea exactă de rerutare automată client, inclusiv dezactivarea, folosind variabilele de registru **DB2_MAX_CLIENT_CONNRETRIES** și **DB2_CONNRETRIES_INTERVAL**. Variabila de registru pentru timeout conexiune este **DB2TCP_CLIENT_CONTIMEOUT**.

Considerente pentru exploatarea System z SYSPLEX

DB2 Connect furnizează echilibrarea încărcării și toleranță la erori la rutarea conexiunilor la mai multe Sysplex-uri. În timpul conexiunii la un server bază de date DB2 for z/OS care rulează într-un mediu de partajare date, DB2 Connect va împrăști încărcarea de lucru între diferitele subsisteme DB2 care conțin grupul de partajare date, pe baza informațiilor de încărcare sistem furnizate de Workload Manager (WLM).

DB2 Connect primește o listă prioritară de membrii Sysplex de la WLM. Fiecare Sysplex returnează informații de prioritate ponderată pentru fiecare adresă de conexiune. Această listă este apoi folosită de DB2 Connect pentru a manipula cererile de intrare CONNECT distribuindu-le între membrii Sysplex care au alocate prioritățile cele mai mari. Pentru echilibrarea încărcării, lista cu informațiile de priorități ponderate Sysplex este obținută în timpul fiecărei conexiuni. Dacă este activat concentratorul de conexiuni DB2 Connect, această listă este de asemenea folosită pentru a se determina unde să fie trimisă fiecare tranzacție.

Notă: Configurația System z Distributed Data Facility (DDF) nu trebuie modificată pentru a profita de exploatarea DB2 Connect Sysplex.

DB2 Connect furnizează de asemenea toleranță la erori încercând să se conecteze la o mașină Sysplex alternativă în cazul eșuării unei conexiuni. O eroare va fi întoarsă aplicației dacă toate conexiunile cunoscute au eșuat.

DB2 Connect Sysplex este proiectat având în vedere gruparea (pooling) agenților. Cu Sysplex activat, DB2 Connect rutează conexiunile la alt membru DDF în cazul în care este pierdută conexiunea la un membru participant la Sysplex. Rerutarea este realizată în funcție de o listă de servere Sysplex.

Cu adăugarea concentratorului, DB2 Connect are acum abilitatea de a echilibra încărcarea de lucru la limitele tranzacției. Concentratorul DB2 Connect trebuie activat pentru ca aceasta să funcționeze.

Exploatarea DB2 Sysplex

Într-un scenariu tipic, un server DB2 Connect (serverul A) ar conversa cu un Sysplex care conține două servere DB2 for z/OS (serverele B și C).

Sysplex server B	Sysplex server C
HOST_NAME=MVSHOST	HOST_NAME=MVSHOST1

Să presupunem că în acest scenariu o aplicație emite acum:

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

Conexiunea la baza de date MVSHOST este stabilă. Deoarece exploatarea Sysplex este activată și pentru serverul DB2 Connect și pentru intrarea director DCS, DB2 for z/OS identifică adresa de rețea la DB2 Connect pentru fiecare participant Sysplex (MVSHOST și MVSHOST1). Protocoale și fluxuri de mesaje DRDA4 sunt folosite pentru a returna aceste informații. O dată ce a fost făcută o conexiune inițială, lista returnată de adrese este pusă în cache pe stația de lucru DB2 Connect. După ce este emis inițial un CONNECT pentru un nod TCP/IP sunt returnate adresele IP.

Informații de prioritate folosite pentru echilibrarea încărcării și toleranța la eroare

Lista de adrese furnizată de DB2 for z/OS include de asemenea informații de prioritate, inclusiv numărul de conexiuni de la fiecare adresă de rețea. Lista este reîmprospătată dacă este făcută o nouă conexiune de către DB2 Connect. Aceste informații suplimentare sunt folosite în scopuri de echilibrarea încărcării, precum și pentru toleranța la erori.

Lista de adrese puse în cache folosită de DB2 Connect

Dacă conexiunea bază de date la ALIASB eșuează, atunci mesajul este eroare SQL30081N este emis și conexiunea va fi abandonată. Dacă mai este primită o cerere ulterioară de conexiune pentru ALIASB, DB2 Connect face următoarele:

1. Încearcă serverul cu cea mai mare prioritate din lista cache de adrese pe baza informațiilor de prioritate care au fost returnate de DB2 for z/OS. Această strategie este mereu folosită de DB2 Connect și astfel se realizează echilibrarea încărcării.
2. Dacă eșuează această încercare de conexiune, atunci celelalte adrese din listă sunt încercate, în ordinea descrescătoare a priorității, precum este returnat de DB2 for z/OS. Acesta este modul în care DB2 Connect exploatează informațiile Sysplex pentru a realiza toleranța erorilor.
3. Dacă toate celelalte cereri de conexiune eșuează, atunci DB2 Connect va reîncerca să se conecteze la ALIASB folosind adresa conținută în directorul de noduri catalogate.

Linia de comandă db2pd cu parametrul sysplex (db2pd -sysplex) poate fi folosită pentru extragerea informațiilor despre servere asociate cu un mediu Sysplex.

Cerințele de configurare pentru Sysplex

Exploatarea Sysplex nu va fi folosită pentru o bază de date decât dacă intrarea director DCS pentru acea bază de date conține Sysplex (insensibil la majuscule) în parametrul din poziția a 6-a.

Capitolul 19. Suportul Sysplex pentru client

Clienții IBM Data Server și driver-ele serverelor de date non-Java care au o licență DB2 Connect pot accesa un DB2 for z/OS Sysplex și pot utiliza toate capabilitățile Sysplex, inclusiv echilibrarea încărcării la nivel de tranzacție, rerutarea automată a clientului și suportul XA limitat.

Un Sysplex este o colecție de sisteme DB2 (numite membri) care formează un grup de partajare de date. Una sau mai multe facilități de cuplare furnizează o memorare în cache de mare viteză și procesarea blocărilor pentru grupul de partajare de date. Sysplex, împreună cu Workload Manager (WLM), adresa IP virtuală dinamică (DVIPA) și cu Sysplex Distributor, permit unui client să acceseze o bază de date DB2 for z/OS peste TCP/IP cu reziliență de rețea și tranzacții distribuite pentru o aplicație, într-o manieră echilibrată între membrii grupului de partajare de date.

Punctul central al acestor capabilități este o listă de servere pe care o întoarce fiecare membru al grupului de partajare de date DB2 la marginile conexiunii și, opțional, la marginile tranzacției. Lista conține adresa IP și capacitatea disponibilă pentru fiecare membru DB2. Cu aceste informații, un client poate distribui tranzacțiile într-o manieră echilibrată sau poate identifica membrul DB2 de folosit când există un eșec de comunicație.

Lista de servere este întoarsă la prima conectare de succes la baza de date DB2. De aceea, conexiunea inițială la baza de date trebuie direcționată la adresa IP virtuală dinamică (DVIPA) grupul deținută de Sysplex Distributor. Dacă cel puțin un membru DB2 este disponibil, Sysplex Distributor va ruta cererea către baza de date. După ce clientul a primit lista de servere, clientul accesează direct un membru DB2 bazat pe informațiile din lista de servere.

Pentru a configura suportul Sysplex, specificați setările în fișierul de configurație db2dsdriver.

Pentru mai multe informații despre WLM, DVIPA și Sysplex Distributor, vedeți DB2 for z/OS Information Center la <http://publib.boulder.ibm.com/infocenter/dzichelp>.

Echilibrarea încărcării de lucru la nivel de tranzacție (pe partea client)

Clienții IBM Data Server și driver-ele serverului de date non-Java care au o licență DB2 Connect pot realiza echilibrarea încărcării de lucru la nivel de tranzacție fără să mai treacă prin serverul DB2 Connect. Cu echilibrarea încărcării de lucru la nivel de tranzacție, încărcarea de lucru pe membri la un DB2 for z/OS Sysplex poate fi reechilibrată la începutul tranzacției.

Echilibrarea încărcării de lucru este realizată pe tranzacții concurente pe o singură conexiune sau pe conexiuni multiple pe o bază per aplicație. Pentru aplicațiile care invocă doar o singură conexiune care rulează un timp îndelungat unde tranzacțiile rulează mai mult consecutiv decât concurent, încărcarea de lucru se mută la membrul cu capacitatea cea mai mare. Algoritmul de echilibrare a încărcării de lucru face automat ajustările pentru a modifica ponderile gestiunii încărcării de lucru.

Următoarea privire generală descrie pașii care apar când un client se conectează la un DB2 for z/OS Sysplex și când apare echilibrarea încărcării de lucru cu re folosirea transportului:

1. Când clientul stabilește o conexiune, serverul întoarce o listă de servere care conține informații de distribuție a încărcării de lucru a membrilor.

Lista de servere este memorată în cache de client și are un timp de viață implicit de 30 de secunde.

2. La începutul unei noi tranzacții, clientul citește lista de servere pentru a identifica un membru care are capacitate nefolosită și caută în pool-ul de transport pentru un transport inactiv care este legat la un membru mai puțin folosit. (Un transport inactiv (idle) este un transport care nu are asociată o conexiune logică.)
 - Dacă este disponibil un transport inactiv, clientul asociază conexiunea logică cu transportul.
 - Dacă, după un timeout configurabil de utilizator, nu este disponibil niciun transport inactiv în pool-ul de transport și nu poate fi alocat niciun transport nou, deoarece pool-ul de transport și-a atins limita, se întoarce o eroare aplicație.
3. Când rulează tranzacția, ea accesează membrul care este legat la transport.
4. Când tranzacția se termină, clientul verifică cu serverul dacă re folosirea transportului mai este permisă pentru conexiunea logică.
5. Dacă re folosirea transportului este permisă, serverul întoarce o listă de instrucțiuni SET pentru registrele speciale care se aplică mediului de execuție pentru conexiunea logică. Clientul memorează în cache aceste instrucțiuni, pe care le redă pentru a reconstrui mediul de execuție când conexiunea logică este asociată unui nou transport.
6. Dacă re folosirea transportului este permisă, conexiunea logică este disociată periodic de transport și copia clientului pentru lista de servere este reîmprospătată.
7. Când echilibrarea încărcării de lucru este necesară pentru o nouă tranzacție, clientul folosește același proces de a asocia conexiunea logică cu un transport.

Indiciu: Echilibrarea încărcării de lucru la nivel de tranzacție pe partea clientului este realizată pe tranzacțiile care rulează pe un singur proces de aplicație. Dacă mai multe procese de aplicație care rulează pe client accesează aceeași bază de date activată pentru Sysplex, echilibrarea încărcării de lucru este realizată pentru fiecare proces de aplicație independent de celelalte procese. Pentru a realiza echilibrarea încărcării de lucru la nivel de tranzacție pe mai multe conexiuni de la mai multe aplicații sau procese care rulează pe mai mulți clienți la distanță, folosiți caracteristicile de echilibrare a încărcării de lucru disponibile la serverul DB2 Connect.

Configurarea echilibrării încărcării de lucru la nivel de tranzacție (pe partea client)

Pentru a configura echilibrarea încărcării de lucru la nivel de tranzacție la client, specificați setările în fișierul de configurare `db2dsdriver`.

Înainte de a începe

Pentru a realiza echilibrarea încărcării de lucru la nivel de tranzacție, clientul trebuie să folosească conexiunea TCP/IP și să aibă o licență DB2 Connect. Următorii clienți furnizează suport pentru echilibrarea încărcării de lucru la nivel de tranzacție:

- IBM Data Server Client
- IBM Data Server Runtime Client
- IBM Data Server Driver Package
- IBM Data Server Driver for ODBC and CLI

Despre acest task

Acest task configurează echilibrarea încărcării de lucru la nivel de tranzacție, ceea ce permite unui client să realizeze echilibrarea încărcării de lucru la nivel de tranzacție fără să treacă printr-un server DB2 Connect.

Procedura

1. În fișierul de configurare `db2dsdriver`, activați echilibrarea încărcării de lucru la nivel de tranzacție prin setarea parametrului `enableWLB` la "true" în subsecțiunea `WLB` a unei intrări de bază de date sau a intrării `DSN`. De exemplu, specificați următorul fișier de configurare `db2dsdriver`.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- parametri specifici bazei de date -->
  <WLB>
    <!-- Sysplex WLB este dezactivat implicit -->
    <parameter name="enableWLB" value="true" />
  </WLB>
</database>
```

Implicit, `enableWLB` este false, iar echilibrarea încărcării de lucru este dezactivată.

2. Opțional: ajustați fin setările pentru echilibrarea încărcării de lucru specificând valorile pentru următorii parametri. Valorile implicite pentru acești parametri ar trebui să fie suficiente pentru majoritatea aplicațiilor.

Tabela 17. Setările echilibrării încărcării de lucru în fișierul de configurație `db2dsdriver`

Parametru	Descriere
<code>maxTransports</code>	Specifică numărul maxim de transporturi în pool-ul de transport. Valoarea implicită este -1 (nelimitat). Orice altă valoare negativă nu este validă. O valoare de 0 dezactivează echilibrarea încărcării de lucru.
<code>maxTransportIdleTime</code>	Specifică timpul maxim scurs în număr de secunde înainte ca un transport inactiv să fie abandonat. Valoarea implicită este 600. Valoarea minimă suportată este 0.
<code>maxTransportWaitTime</code>	Specifică numărul de secunde pe care le va aștepta clientul până când un transport devine disponibil. Valoarea implicită este -1 (nelimitat). Valoarea minimă suportată este 0.
<code>maxRefreshInterval</code>	Specifică timpul maxim scurs în număr de secunde înainte ca lista de servere să fie reimprospătată. Valoarea implicită este 30. Valoarea minimă suportată este 0.

Pentru mai multe informații, vedeți subiectul despre fișierul de configurare `db2dsdriver`, în Centrul de informare `DB2`.

Notă: Setările de configurație existente ale managerului de bază de date `MAX_COORDAGENTS`, `NUM_POOLAGENTS` și `MAX_CONNECTIONS` sunt folosite pentru configurarea echilibrării încărcării de lucru la serverul `DB2 Connect` și de aceea nu au niciun efect asupra echilibrării încărcării de lucru realizate pe partea clientului.

Exemplu

Exemplu 1: Activarea echilibrării încărcării de lucru la nivel de tranzacție pentru baza de date `SAMPLE` folosind setările implicite.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- parametri specifici bazei de date -->
  <WLB>
    <!-- Sysplex WLB este dezactivat implicit -->
    <parameter name="enableWLB" value="true" />
  </WLB>
</database>
```

Exemplu 2: Activarea echilibrării încărcării de lucru la nivel de tranzacție pentru baza de date SAMPLE cu un maximum de 95 pentru transporturi.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- parametri specifici bazei de date -->
  <WLB>
    <!-- Sysplex WLB este dezactivat implicit -->
    <parameter name="enableWLB" value="true" />
    <!-- maxTransports reprezintă numărul maxim de transporturi;
    implicit este -1 (nelimitat) -->
    <parameter name="maxTransports" value="95" />
  </WLB>
</database>
```

Exemplu 3: Activarea echilibrării încărcării de lucru la nivel de tranzacție pentru baza de date SAMPLE cu un maximum de 80 pentru transporturi și un timp maxim de așteptare de 10 secunde pentru un transport disponibil.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- parametri specifici bazei de date -->
  <WLB>
    <!-- Sysplex WLB este dezactivat implicit -->
    <parameter name="enableWLB" value="true" />
    <!-- maxTransports reprezintă numărul maxim de transporturi;
    implicit este -1 (nelimitat) -->
    <parameter name="maxTransports" value="80" />
    <!-- maxTransportWaitTime reprezintă numărul maxim de timp scurs în;
    secunde, înainte ca un transport să devină disponibil;
    implicit este -1 (nelimitat) -->
    <parameter name="maxTransportWaitTime" value="10" />
  </WLB>
</database>
```

Exemplu 4: Dezactivarea echilibrării încărcării de lucru la nivel de tranzacție pentru baza de date SAMPLE.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- parametri specifici bazei de date -->
  <WLB>
    <!-- Sysplex WLB este deja dezactivat implicit -->
    <parameter name="enableWLB" value="false" />
  </WLB>
</database>
```

Rerutarea automată a clientului (pe partea client)

Caracteristica ACR (Automatic Client Reroute) furnizează suport pentru preluarea la defect atunci când un client IBM Data Server își pierde conectivitatea la un membru DB2 for z/OS Sysplex. ACR permite clientului să recupereze dintr-un defect prin încercarea de reconectare la baza de date prin intermediul oricărui membru Sysplex disponibil.

ACR este activat implicit când este activată echilibrarea de încărcare de lucru.

Suportul ACR pe partea clientului este disponibil în clienții IBM Data Server și în driver-ele serverelor de date non-Java care au o licență DB2 Connect. Serverul DB2 Connect nu este necesar pentru a realiza rutarea automată client (ACR).

Dacă ACR este activat, când un client întâmpină un defect de conexiune la o conexiune existentă, în mod obișnuit se întâmplă următorul proces:

1. Clientul încearcă să execute o instrucțiune SQL folosind o conexiune existentă și întâlnește un defect.
2. Clientul folosește lista de servere pentru a identifica membrul Sysplex de accesat și încearcă din nou să se conecteze la baza de date. Această listă de servere poate fi lista

returnată de server, sau, dacă afinitățile clientului sunt specificate, o listă de servere alternative specificată de client. Pentru informații suplimentare, vedeți “Configurarea afinităților clientului” la pagina 114.

3. Dacă ACR nu poate să se reconecteze la baza de date, clientul reconstruiește mediul de execuție pentru noua conexiune stabilită. Este returnată aplicației eroarea SQL30108N pentru a indica recuperarea conexiunii la baza de date eșuată și că tranzacția fost derulată înapoi. Aplicația este responsabilă mai departe pentru recuperare, inclusiv de repetarea tuturor lucrărilor care au fost derulate înapoi.
4. Dacă ACR nu poate să se reconecteze la baza de date, eroarea SQL30081N este returnată aplicației. Aplicația este apoi responsabilă pentru recuperarea din eșecul conexiunii (de exemplu, încercând ea însăși să se conecteze la baza de date).

ACR este folosită și când un client întâlnește un defect de conexiune cu o conexiune nouă. În acest caz, nu este returnată aplicației eroarea SQL30108N pentru a indica recuperarea conexiunii la baza de date eșuată. Fie conectarea are succes, fie este întoarsă eroarea SQL30081N.

Puteți dezactiva ACR sau să configurați mai în detaliu rutarea prin specificarea setărilor în fișierul de configurație db2dsdriver.

Preluarea la defect de tip seamless pentru aplicațiile CLI și .NET

Când ACR este activat, iar destinația tranzacției este DB2 for z/OS, preluarea la defect de tip seamless pentru aplicațiile CLI și .NET este activată implicit. Cu preluarea la defect de tip seamless, dacă aplicația întâlnește un defect de conexiune la prima operație SQL într-o tranzacție, driver-ul rulează din nou operația SQL eșuată ca parte a procesării rutării automate client. Dacă conectarea are succes, nu se raportează nicio eroare aplicației, iar tranzacția nu va fi derulată înapoi. Eșecul conectivității și recuperarea ulterioară sunt ascunse aplicației.

Puteți dezactiva sau reactiva preluarea la defect de tip seamless prin specificarea setărilor în fișierul de configurare db2dsdriver.

Considerațiile privind serverul DB2 Connect legate de ACR și preluarea la defect de tip seamless

Comportamentul ACR și al preluării la defect de tip seamless este diferit dacă clientul accesează Sysplex-ul prin intermediul unui server DB2 Connect.

Dacă eroarea apare între serverul DB2 Connect și Sysplex, rutarea automată client (ACR) este realizată de serverul DB2 Connect. Dacă serverul DB2 Connect este la același nivel cu client sau mai sus, clientul poate realiza preluarea la defect de tip seamless (fără cusături). Altfel, clientul nu poate realiza preluarea la defect de tip seamless și este returnată aplicației eroarea SQL30108N pentru a indica recuperarea conexiunii eșuate la baza de date.

Dacă eroarea apare între client și serverul DB2 Connect și Sysplex, rutarea automată client (ACR) poate fi realizată pe client la serverul DB2 Connect. Dar preluarea la defect de tip seamless este întotdeauna, iar eroarea SQL30108N este întoarsă aplicației.

Configurarea rerutării automate a clientului (pe partea client)

Rerutarea automată a clientului (ACR) este activată implicit când este activată echilibrarea încărcării de lucru. Puteți dezactiva ACR sau să configurați mai în detaliu rutarea prin specificarea setărilor în fișierul de configurație db2dsdriver.

Înainte de a începe

Pentru a realiza ACR, clientul trebuie să folosească conexiunea TCP/IP și să aibă o licență DB2 Connect. Următorii clienți furnizează suport pentru ACR:

- IBM Data Server Client
- IBM Data Server Runtime Client
- IBM Data Server Driver Package
- IBM Data Server Driver for ODBC and CLI

Despre acest task

Pentru unele aplicații, ați putea dori să dezactivați ACR sau preluarea la defect de tip seamless, sau ați putea dori să configurați ACR mai în detaliu. Acest task descrie parametrii care sunt disponibili pentru configurarea ACR.

Puteți de asemenea să modificați comportamentul ACR specificând afinitățile clientului pentru a controla lista de servere care este folosită pentru procesarea conectării și pentru ACR. Pentru informații, vedeți “Configurarea afinităților clientului” la pagina 114.

Procedura

Pentru a dezactiva ACR sau pentru a configura mai în detaliu rerutarea:

- Pentru a dezactiva ACR sau ACR cu preluare la defect seamless, specificați o valoare de “false” pentru următorii parametri în fișierul de configurare `db2dsdriver`. Specificați aceste setări în subsecțiunea ACR într-o intrare de bază de date.

Tabela 18. Setările rutării automate a clientului în fișierul de configurație db2dsdriver

Parametru	Descriere
<code>enableAcr</code>	Specifică dacă este activată (true) caracteristica ACR sau nu (false). Această setare este implicit true când echilibrarea încărcării de lucru la nivel de tranzacție este activată (setarea <code>enableWLB</code> este true). Dacă ACR este dezactivat, celelalte setări pentru ACR sunt ignorate.
<code>enableSeamlessAcr</code>	Specifică dacă ACR cu caracteristica de preluare la defect de tip seamless este activată (true) sau nu (false). Când ACR este activat, iar serverul destinație este DB2 for z/OS, preluarea la defect de tip seamless este activată implicit. Dacă serverul destinație nu este DB2 for z/OS, această setare este ignorată, iar ACR seamless nu este suportat.

Pentru mai multe informații, vedeți subiectul despre fișierul de configurare `db2dsdriver`, în Centrul de informare DB2.

- Puteți specifica o listă de servere alternative care vor fi folosite dacă apare un defect la prima încercare de conectare, iar gazda specificată pentru conexiune nu este disponibilă. Pentru a specifica o listă de servere alternative, specificați o valoare de “true” pentru `enableAlternateServerListFirstConnect` în subsecțiunea ACR din intrarea pentru baza de date. Când valoarea pentru `enableAlternateServerListFirstConnect` este “true,” ACR cu preluare la defect seamless este activat implicit indiferent de celelalte setări care sunt specificate pentru ACR în fișierul de configurare `db2dsdriver`. Pentru a folosi această caracteristică, trebuie să setați și o listă de servere alternative în fișierul de configurare `db2dsdriver`.

Pentru informații despre setarea unei liste de servere alternative, vedeți “Configurarea afinităților clientului” la pagina 114. Deși informațiile despre setarea unei liste de servere

alternative apar în subiectul care descrie afinitățile clientului, nu trebuie să folosiți afinitățile clientului pentru a face uz de această capacitate. De fapt, această capacitate este suportată când este activată echilibrarea încărcării de lucru la nivel de tranzacție, chiar dacă echilibrarea încărcării de lucru la nivel de tranzacție și afinitățile client nu pot fi activate în același timp.

Rezultate

Dacă ACR este activat, o aplicație care întâlnește un eșec al conectivității este rutată la un alt membru Sysplex. Dacă preluarea la defect de tip seamless este dezactivată sau ne se poate aplica, eroarea SQL30108N este întoarsă aplicației.

Dacă defectul apare la prima operație SQL a unei tranzacții, iar preluarea la defect de tip seamless este activată (și aplicabilă), clientul poate rula din nou operația SQL eșuată ca parte a procesării ACR. Dacă conectarea are succes, nu se raportează nicio eroare aplicației.

Exemplu

Exemplu 1: Activare ACR pentru baza de date SAMPLE când preluarea la defect de tip seamless este dezactivată

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- parametri specifici bazei de date -->
  <WLB>
    <!-- Sysplex WLB este dezactivat implicit -->
    <parameter name="enableWLB" value="true" />
  </WLB>
  <ACR>
    <!-- ACR este deja activat implicit când WLB este activat -->
    <parameter name="enableAcr" value="true" />
    <!-- Când ACR este activat, iar serverul destinație este DB2 for z/OS,
    Seamless ACR este activat implicit; dacă serverul destinație nu este
    DB2 for z/OS, această setare este ignorată iar Seamless ACR
    nu este suportat. -->
    <parameter name="enableSeamlessAcr" value="false" />
  </ACR>
</database>
```

Exemplu 2: Dezactivare ACR pentru baza de date SAMPLE

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- parametri specifici bazei de date -->
  <WLB>
    <!-- Sysplex WLB este dezactivat implicit -->
    <parameter name="enableWLB" value="true" />
  </WLB>
  <ACR>
    <!-- ACR este activat implicit când WLB este activat -->
    <parameter name="enableAcr" value="false" />
  </ACR>
</database>
```

Exemplu 3: Specificați că lista de servere alternative este folosită la momentul primei conectări, dacă gazda inițială nu este disponibilă

```
<database name="SAMPLE" host="v33ec065.svl.ibm.com" port="446">
  <ACR>
    <parameter name="enableACR" value="true"/>
    <parameter name="enableAlternateServerListFirstConnect"
    value="true"/>
  </ACR>
</database>
```

Configurarea afinităților clientului

Puteți configura afinitățile clientului pentru a controla ordinea serverelor care sunt încercate în timpul procesării conectării inițiale și în timpul procesării rerutării automate a clientului (ACR). Puteți folosi afinitățile clientului când echilibrarea încărcării de lucru nu este activată explicit.

Despre acest task

În mod obișnuit, aplicațiile care folosesc capacitățile Sysplex ale clientului determină la care membru să se conecteze în timpul unui defect, folosind cea mai recentă listă de servere returnată de serverul de bază de date. În acest model, clientul rerutează pe baza ultimei liste de servere returnate pe care a citit-o, prin urmare ordinea serverelor de încercat este controlată de server. Totuși, puteți neglija această comportare configurând afinitățile clientului și specificând o listă de servere alternativă de folosit în timpul procesării conectării inițiale și pentru procesarea rerutării clientului realizate de client. Când procesarea afinităților clientului este activă, ordinea serverelor care sunt încercate pentru procesarea ACR este controlată de către client.

Pentru a configura afinitățile clientului, specificați setări pentru `ALTERNATE_SERVER_LIST`, `AFFINITY_LIST` și `CLIENT_AFFINITY` în fișierul de configurare `db2dsdriver`. Când aceste setări sunt specificate:

- Conexiunea este stabilită pe baza unei liste de servere alternativă, nu pe baza portului și gazdei specificate de aplicație. De exemplu, dacă aplicația specifică `Host = s1` și lista de servere alternativă specifică `Serverlist = (s2, s1, s3)`, atunci conexiunea inițială este încercată pe `s2` și nu pe `s1`. Ordinea încercărilor de conectare este `(s2,s2,s2)` `(s1,s1,s1)` `(s3,s3,s3)`. `acrRetryInterval` se aplică încercărilor individuale de conectare (un total de 8 ori, în acest exemplu).
- Rerutarea clientului de tip `seamless` (fără cusături) este activată automat. Rerutarea clientului este realizată prin conectarea la serverele specificate în parametrii `Server` și `Port` ai grupului `ALTERNATE_SERVER_LIST`. Fiecare server din lista alternativă este încercat de `maxAcrRetries` ori înainte de a fi folosit următorul server din lista alternativă, iar aceasta continuă până când este parcursă întreaga listă sau până se stabilește cu succes o conexiune. De exemplu, dacă lista de servere alternativă este definită să fie `(s3,s1,s2)` și `maxAcrRetries` este 3, atunci ordinea de rerutare este `(s3,s3,s3)` `(s1,s1,s1)` `(s2,s2,s2)`. `acrRetryInterval` se aplică după fiecare încercare individuală de conectare (un total de 8 ori, în acest exemplu).

Procedura

Pentru a specifica setările de afinitate client în fișierul de configurare `db2dsdriver`.

1. În fișierul de configurare `db2dsdriver` sub grupul `ACR`, declarați lista de servere alternative specificând intrări pentru `ALTERNATE_SERVER_LIST`. În listă specificați numărul de port și numele de gazdă pentru fiecare server. De exemplu:

```
<database name="SAMPLE" host="v33ec065.svl.ibm.com" port="446">
  <acr>
    <parameter name="enableACR" value="true"/>
    <alternate_server_list>
      <server name="server1"
        hostname="v33ec067.svl.ibm.com"
        port="446" >
    </server>
      <server name="server2"
        hostname="v33ec066.svl.ibm.com"
        port="446" >
    </server>
      <server name="server3"
```

```

        hostname="v33ec065.svl.ibm.com"
        port="446" >
    </server>
</alternate_server_list>
.
.
.

```

Specificând această listă, doar declară serverele alternative; nu duce la nicio modificare în comportament.

- Sub grupul ACR, identificați ordinea serverelor specificând intrări în AFFINITY_LIST. De exemplu:

```

.
.
.
<affinity_list>
  <list name="list1"
    serverorder="server1,server2,server3" >
  </list>
  <list name="list2"
    serverorder="server3,server2,server1" >
  </list>
</affinity_list>
.
.
.

```

Specificând această listă, doar identifică ordinea serverelor; nu duce la nicio modificare în comportament.

- Sub grupul ACR, activați afinitățile clientului specificând unul din următoarele subgrupuri de afinități client. Specificând unul dintre aceste subgrupuri se forțează activarea afinităților clientului. Toți clienții care se conectează la această bază de date trebuie specificați în unul din subgrupurile CLIENT_AFFINITY. Dacă un client nu este găsit în unul din aceste subgrupuri, atunci se va ridica o eroare la încercarea de conectare. Când este prezent un subgrup CLIENT_AFFINITY, ACR este activat implicit.

- CLIENT_AFFINITY_DEFINED

Specifică o anumită mapare de la numele de gazdă client la un element AFFINITY_LIST specific. Numele de gazdă client va fi descoperit automat și va fi comparat cu intrarea din fișierul de configurație db2dsdriver pentru a calcula lista de afinități. De exemplu:

```

.
.
.
<client_affinity_defined>
  <!-- această secțiune are afinități specifice definite -->
  <client name="client1"
    hostname="appsrv1.svl.ibm.com"
    listname="list2" >
  </client>
  <client name="client2"
    hostname="appsrv2.svl.ibm.com"
    listname="list1" >
  </client>
</client_affinity_defined>
.
.
.

```

- CLIENT_AFFINITY_ROUNDROBIN

Specifică o alocare circulară (round robin) în ALTERNATE_SERVER_LIST. Această alocare are serverul de început setat la fel cu indexul clientului (baza zero) în lista

CLIENT_AFFINITY_ROUNDROBIN, modulo numărul de servere din ALTERNATE_SERVER_LIST. De exemplu:

```
.
.
.
<client_affinity_roundrobin>
  <!-- roundrobin alege serverul de început la fel cu
        numărul de index al clientului din această secțiune (baza 0)
        modulo numărul de servere.
  -->
  <client name="client3"
        hostname="appsrv3.svl.ibm.com" >
    <!-- această intrare este index 0, modulo 3, așa că obține:
        server1, server2, server3
    -->
  </client>
  <client name="client4"
        hostname="appsrv4.svl.ibm.com" >
    <!-- această intrare este index 1, modulo 3, așa că obține:
        server2, server3, server1
    -->
  </client>
</client_affinity_roundrobin>
.
.
.
```

Rezultate

La procesarea afinităților clientului se aplică următoarele reguli și considerente:

- Când este activat CLIENT_AFFINITY (adică, sunt prezente unul sau mai multe subgrupuri CLIENT_AFFINITY în fișierul de configurare db2dsdriver), valoarea implicită pentru maxAcrRetries este 3. Intervalul valid pentru această valoare este de la 0 la MAX_INT. Dacă valoarea specificată este 0, procesarea ACR se realizează ca și cum maxAcrRetries ar fi fost specificat la 1.
- Valoarea implicită pentru acrRetryInterval este 0. Intervalul valid pentru această valoare este de la 0 la MAX_INT. Când această valoare este 0, nu există niciun interval între reîncercări sau încercările ACR în driver.
- Numărul total de servere alternative care se califică pentru un client dat nu poate depăși 24. Când numărul de servere care se califică este mai mare decât 24, apare SQL1042N.
- Setările enableWLB și CLIENT_AFFINITY se exclud reciproc. Apare o eroare SQL5162N dacă sunt activate amândouă setările enableWLB și CLIENT_AFFINITY.
- Atributele necesare trebuie specificate în grupurile ALTERNATE_SERVER_LIST, AFFINITY_LIST și CLIENT_AFFINITY. Altfel va apare SQL5163N.
- Dacă este activat CLIENT_AFFINITY, ALTERNATE_SERVER_LIST nu trebuie să fie goală. Altfel va apare SQL5164N.
- Dacă este activată CLIENT_AFFINITY, atunci numele de gazdă ai clienților trebuie să se rezolve la o listă de servere alternative validă. Altfel va apare SQL5164N. De exemplu, dacă numele de gazdă al clientului este "xyz.domain.com" și numele de gazdă nu este găsit în unul din subgrupurile CLIENT_AFFINITY, atunci apare SQL5164N. În plus, numele de gazdă poate apărea doar în unul din grupurile CLIENT_AFFINITY.

Exemplu

Următorul exemplu arată o secțiune de bază de date exemplu a fișierului de configurare db2dsdriver cu subgrupurile CLIENT_AFFINITY specificate:

```

<database name="SAMPLE" host="v33ec065.svl.ibm.com" port="446">
  <acr>
    <parameter name="enableACR" value="true"/>
    <alternate_server_list>
      <server name="server1"
        hostname="v33ec067.svl.ibm.com"
        port="446" >
      </server>
      <server name="server2"
        hostname="v33ec066.svl.ibm.com"
        port="446" >
      </server>
      <server name="server3"
        hostname="v33ec065.svl.ibm.com"
        port="446" >
      </server>
    </alternate_server_list>
    <affinity_list>
      <list name="list1"
        serverorder="server1,server2,server3" >
      </list>
      <list name="list2"
        serverorder="server3,server2,server1" >
      </list>
    </affinity_list>
    <client_affinity_defined>
      <!-- this section has specific defined affinities
      -->
      <client name="client1"
        hostname="appsrv1.svl.ibm.com"
        listname="list2" >
      </client>
      <client name="client2"
        hostname="appsrv2.svl.ibm.com"
        listname="list1" >
      </client>
    </client_affinity_defined>
    <client_affinity_roundrobin>
      <!-- roundrobin alege serverul de început la fel cu
        numărul de index al clientului din această secțiune (baza 0)
        modulo numărul de servere.
      -->
      <client name="client3"
        hostname="appsrv3.svl.ibm.com" >
        <!-- această intrare este index 0, modulo 3, așa că obține:
        server1, server2, server3
      -->
      </client>
      <client name="client4"
        hostname="appsrv4.svl.ibm.com" >
        <!-- această intrare este index 1, modulo 3, așa că obține:
        server2, server3, server1
      -->
      </client>
    </client_affinity_roundrobin>
  </acr>
</database>

```

Limitări la folosirea suportului de Sysplex pentru client

Se aplică unele limitări suportului Sysplex pentru client disponibil în clienții IBM Data Server și driver-ele de server de date non-Java.

- Limitări ale echilibrării încărcării de lucru la nivel de tranzacție:
 - Clientul trebuie să folosească o conexiune TCP/IP.

- Pentru clienții și driver-ele IBM Data Server care folosesc suportul Sysplex pentru client, lista de servere este pusă în cache în cadrul procesului aplicației. De aceea, lista este partajată doar de conexiunile din acel proces pentru echilibrarea încărcării de lucru de la unul sau mai multe fire de execuție.
- Limitările rerutării automate a clientului:
 - Clientul trebuie să folosească o conexiune TCP/IP.
 - Preluarea la defect de tip seamless (fără cusături) la prima instrucțiune SQL a unei tranzații nu este suportată dacă serverul Sysplex nu permite re folosirea transportului la sfârșitul tranzației anterioare. O excepție este dacă re folosirea transportului nu este acordată ca rezultatul unui KEEP DYNAMIC(YES).
 - Preluarea la defect de tip seamless la prima instrucțiune SQL a unei tranzații nu este suportată dacă driver-ul CLI trebuie să păstreze o istorie a API-urilor apelate anterior pentru a putea reda instrucțiunea SQL. Scenariile în care aceasta poate apărea includ specificarea datelor la momentul execuției, la efectuarea de SQL compus și la folosirea intrărilor matriciale.
 - Dacă se întâmplă o preluare la defect de tip seamless când autocomiterea este activată, o instrucțiune SQL poate fi executată și comisă de mai multe ori. Acest rezultat poate apărea când apare un eveniment care declanșează rerutarea automată a clientului după terminarea cu succes a comiterii originale. Pentru a evita acest comportament, dezactivați fie autocomiterea, fie preluarea la defect de tip seamless (fără cusături).
 - Preluarea la defect de tip seamless nu este aplicabilă în interiorul unei proceduri memorate
 - Preluarea la defect de tip seamless nu este aplicabilă în mediile Federated Server
- Suport XA:
 - Următoarele modele de procesare a managerului de tranzații nu sunt suportate:
 - Dual-transport. În acest model, o tranzație, peste transportul A, este legată la un membru de la xa_start până la xa_end, dar xa_prepare(readonly), xa_prepare plus xa_commit sau xa_rollback, sau xa_rollback vin peste transportul B, posibil din alt proces de aplicație. Exemple de manageri de tranzație care folosesc acest model sunt IBM WebSphere MQ și IBM Lotus Domino.
 - Multi-transport. Acest model implică folosirea mai multor transporturi de la mai multe procese de aplicație, pentru aceeași tranzație. Un exemplu de manager de tranzații care folosește acest model este BEA Tuxedo.
 - Pentru managerii de tranzații XA care folosesc un model de procesare multi-transport, mai este necesar un server DB2 Connect de nivel intermediar (middle-tier).

Capitolul 20. Ajustarea DB2 Connect

Pot fi utilizați diverși parametri din fișierul de configurare al managerului bazei de date pentru a regla DB2 Connect.

RQRIOBLK

Parametrul **RQRIOBLK** setează dimensiunea maximă a blocurilor I/E ale rețelei. O dimensiune mai mare de bloc ar putea îmbunătăți performanța cererilor mai mari. Dimensiunea blocului nu afectează în mod normal timpul de răspuns pentru cereri mici, cum ar fi cererile pentru o singură linie de date.

O dimensiune mai mare a blocului necesită de obicei mai multă memorie pe serverul DB2 Connect. Aceasta crește dimensiunea setului de lucru și ar putea produce un volum mare de paginare pe stațiile de lucru mici.

Folosiți dimensiunea de bloc DRDA implicită (32767) dacă nu cauzează prea multă paginare la executarea aplicației dumneavoastră. Altfel, reduceți dimensiunea blocului de I/E până nu mai există paginare. O dată paginarea pornită, se va produce o degradare a performanței. Folosiți unelte de monitorizare performanță (cum ar fi unealta vmstat pentru sisteme de operare Linux și UNIX) pentru a determina dacă paginarea are loc pe sistemul dumneavoastră.

DIR_CACHE

Parametrul **DIR_CACHE** determină dacă se pun în cache informațiile directorului. Când se folosește punerea în cache (**DIR_CACHE=YES**), fișierele din director sunt citite și trecute în memoria cache pentru a minimiza regia de creare a structurii de directoare interne și de citire a fișierelor director de fiecare dată când este stabilită o conexiune.

Dacă nu se folosește punerea în cache (**DIR_CACHE=NO**), de fiecare dată când vă conectați la o bază de date directorul corespunzător este citit de pe un disc și apoi se execută căutarea. După ce sunt găsite intrările cerute, toată memoria folosită la căutarea în directoare este eliberată.

În cazul punerii în cache, în timpul procesării db2start este construit un cache de director partajat, care este eliberat când DB2 se oprește. Acest cache este folosit de toate procesele server DB2 (db2agent). De asemenea, este construit un director de cache de aplicație privată când o aplicație emite prima conectare a ei la baza de date și este eliberat când se termină aplicația.

Fiecare cache conține o imagine a directorului de bază de date de sistem, a directorului de servicii de conectare la baza de date și a directorului de noduri. Cache-ul reduce costurile conectării prin eliminarea fișierelor directoare I/E și prin minimizarea căutării de directoare.

Dacă este actualizat un director pus în cache, schimbările nu sunt propagate imediat în celelalte cache-uri. Dacă intrarea directorului nu este găsită în cache, se caută în directorul original.

Punerea în cache crește memoria privată necesară pe durata de viață a aplicației. Fără punere în cache, această memorie este necesară doar când este procesată o căutare de director. Folosirea pe ansamblu a memoriei partajate de DB2 crește ușor, deoarece informațiile directorului care sunt partajate între agenții bazei de date sunt mutate în memoria partajată.

Dimensiunea memoriei necesare pentru cache depinde de numărul intrărilor definite în fiecare director.

NUMDB

Comportamentul DB2 Connect nu era afectat de parametrul de configurare **NUMDB** în versiunile anterioare, dar acest lucru s-a schimbat odată cu Versiunea 8. Acest parametru indică numărul maxim de baze de date la care se pot conecta clienții prin serverul DB2 Connect. Mai precis, numărul maxim de aliasuri de bază de date diferite care pot fi catalogate pe serverul DB2 Connect.

Alți parametri DB2 Connect

AGENTPRI și **MAXAGENTS** sunt depreciați în Versiunea 9.5

Comenzile pentru actualizarea valorii pentru **MAXAGENTS** vor continua să funcționeze, așa că aplicațiile existente nu vor suferi, dar valorile vor fi ignorate. Numele de parametru nu va apărea în nicio listă de configurare. În trecut, numărul total permis de agenți pentru a fi creați pe o partiție DB2 dată era controlat prin parametrul de configurare **MAXAGENTS**. Acum există posibilitatea să se configureze automat numărul de agenți.

Implicit, **NUM_POOLAGENTS** va fi setat la **AUTOMATIC** cu o valoare de 100. De asemenea, **MAX_COORDAGENTS** va fi setat la **AUTOMATIC** cu o valoare implicită de 200.

Pentru a trimite șiruri de contabilizare de la aplicațiile dumneavoastră client la serverul DB2 Connect, folosiți mijloacele specifice API pentru setarea informațiilor de contorizare. Mijloacele specifice API sunt mai rapide decât setarea variabilei de mediu **DB2ACCOUNT**.

IBM Data Server Driver for JDBC and SQLJ

Proprietatea `com.ibm.db2.jcc.DB2BaseDataSource.clientAccountingInformation`

IBM Data Server Provider for .NET

Proprietatea `DB2Connection.ClientAccountingInformation`

CLI/ODBC

Cuvântul cheie configurare `CLI/ODBC ClientAcctStr`

SQL încorporat (C, C++ și COBOL)

Funcția `sqlsact`

Dacă nu aveți nevoie de un fișier de mapare **SQLCODE** ajustat, puteți să îmbunătățiți performanța utilizând maparea **SQLCODE** implicită sau dezactivând maparea **SQLCODE**. Fișierul de mapare implicit este înglobat în biblioteca DB2 Connect; un fișier de mapare ajustat trebuie să fie citit de pe disc, ceea ce afectează performanța.

Ajustarea bazei de date gazdă

Performanța sistemului va fi afectată de performanța serverului bază de date mainframe IBM. Sistemele diferite de gestionare a bazelor de date au caracteristici de performanță diferite. Optimizatorii SQL de pe sisteme diferite, de exemplu, s-ar putea comporta diferit cu aceeași aplicație. Verificați documentația performanței sistemului server bază de date mainframe IBM pentru informații suplimentare.

Ați putea îmbunătăți performanța folosind opțiunile de legare UR (uncommitted read - citire necomisă) sau NC (no commit - fără comitere), dacă sunt disponibile, pentru a evita jurnalizarea.

Notă: Când se folosește UR, datele nejournalizate pot fi numai citite, nu actualizate, și apoi numai dacă gruparea pe blocuri este setată la ALL.

În funcție de serverul de aplicații și granularitatea blocării pe care o furnizează, nivelul de izolare folosit pentru o interogare sau aplicație ar putea avea un efect semnificativ asupra performanței. Baza de date ar trebui să aibă un nivel corespunzător de normalizare, utilizare efectivă a indecșilor și alocare a spațiului bazei de date. Performanța poate fi afectată de asemenea de tipurile de date pe care le utilizați, așa cum se arată în secțiunea următoare.

Considerente ale ajustării rețelei

Cea mai bună cale de a îmbunătăți performanțele generale într-un mediu de baze de date distribuite este de a elimina întârzierile din rețea. Este ceva obișnuit pentru administratorii de rețea să considere o rețea mai eficientă în cazul în care colectează cât mai multe date posibile între transmisii. Această metodă nu este valabilă pentru aplicații cum ar fi bazele de date distribuite, deoarece determină întârzieri în rețea. Utilizatorul final nu vede eficiența rețelei, ci doar întârzierile.

Cele mai multe dispozitive de rețea au parametri de întârziere, iar cele mai multe dintre ele au valori implicite care sunt foarte proaste pentru baze de date distribuite. Pentru îmbunătățirea performanței trebuie să localizați acești parametri și, dacă este posibil, să-i setați la zero. În plus ar trebui să vă asigurați că dimensiunea buffer-ului pentru dispozitiv este destul de mare pentru a preveni retransmisia din cauza datelor pierdute. De exemplu, sistemele UNIX au tipic o adâncime implicită a cozii de transmisie sau recepție de 32. Pentru rezultate mai bune, setați adâncimea cozii la 150. Un parametru corespondent la setările DLC este adâncimea la recepție, care ar trebui să fie de asemenea 150.

Parametrul IOBUF este setat la o valoare prea mică în majoritatea locațiilor. De obicei este setat la 500, dar experiența a arătat că soluția optimă este să fie setat la valoarea 3992 dacă mutați cantități mari de date, în special pentru conexiunile canal, cum ar fi ESCON sau 3172.

Pe un sistem LAN dimensiunile ferestrelor de transmisie și recepție DLC sau LLC pot avea efecte dramatice asupra performanțelor. Valoarea de transmisie trebuie setată la 7 sau mai mult, iar pentru majoritatea configurațiilor cea mai bună valoare de recepție este 4 sau mai puțin.

Dacă rulați Ethernet, ar trebui să setați dimensiunea segmentului TCP la 1500 octeți. Pe o rețea Token-ring sau FDDI această valoare ar trebui să fie 4400 octeți și dacă folosiți un adaptor ESCON cu TCP/IP, dimensiunea segmentului ar trebui să fie întotdeauna 4096.

În sfârșit, pentru rețele TCP/IP, dimensiunea buffer-ului de transmisie și recepție TCP ar trebui setat mai mare ca 32768. Valoarea de 65536 este în general cea mai bună.

Notă: Stabilirea unei conexiuni de la gateway la server (conexiune de ieșire) este mult mai scumpă decât stabilirea unei conexiuni de la client la gateway (conexiune de intrare). Într-un mediu unde mii de clienți se conectează la și deconectează de la server frecvent prin gateway, o cantitate substanțială a timpului de procesare este petrecut stabilind conexiuni de ieșire. DB2 Connect asigură conexiuni pool de-a lungul TCP/IP. Când un client cere deconectarea de la server, gateway-ul abandonează conexiunea de intrare cu clientul, dar păstrează conexiunea de ieșire cu serverul într-un pool. Când un client nou intră în gateway pentru a cere o conexiune, gateway-ul asigură una existentă din pool pentru a reduce timpul pentru conectarea generală și pentru a salva costul maxim de conectare la server CPU.

În Tabela 19 la pagina 122 puteți vedea un sumar al metodelor de ajustare a performanței rețelei.

Tabela 19. Metode de ajustare a performanței rețelei

Ce se caută	Exemplu	Setare	Note
Întârzieri deliberate	Parametri de întârziere pe dispozitivele de rețea	Setare la 0.	Implicit sunt de obicei mai mari.
Buffer-e	Parametrul IOBUF	Setare până la 3992.	Util mai ales pentru ESCON sau alt adaptor canal.
Buffer-e	RUSIZE	Dimensiunea optimă este 4096.	Setarea RUSIZE și RQRIOBLK la aceeași dimensiune ar putea da performanță maximă.
Buffer-e	Ritm	VPACING, PACING și Mode Profiles ar trebui setate la 63.	Utilizați un ritm adaptabil, unde se poate aplica.
Setare adaptor	Adâncimea cozii de transmisie/recepție	Valoarea recomandată este 150.	Implicit este de obicei 32.
Setare TCP	Dimensiune segment	1500 la Ethernet, 4400 la Token-Ring și FDDI.	Adaptoarele ESCON folosite pentru TCP/IP ar trebui setate întotdeauna la 4096.
Setare TCP	Dimensiunea spațiului de emisie/recepție	Trebuie să fie 64K pentru ambele.	Implicit este doar 8192 pentru Windows. Poate fi setat în registrul Windows.

Conflictul pentru resursele de sistem

Performanța poate fi degradată dacă mai multe taskuri din sistem intră în conflict pentru resursele de sistem. Luați în considerare următoarele întrebări:

- Este CPU saturat? Luați în considerare modernizarea sistemului, reducerea încărcării de lucru a sistemului și ajustarea sistemului pentru a reduce regia de procesare.
- Este memoria supra-comisă? Luați în considerare modernizarea memoriei, reducând încărcarea de lucru a sistemului și reglând sistemul pentru a reduce setul de lucru al memoriei.
- Este adaptorul/controlerul de comunicații prea ocupat? Luați în considerare modernizarea rețelei sau împerecherea plăcilor Token-Ring.
- Este unul dintre subsisteme prea ocupat și este acest subsistem în calea datelor?
- Există procese sau taskuri care nu sunt necesare și rulează în sistem? Regula generală este să nu se configureze sau pornească servicii decât dacă sunt utilizate cu regularitate pentru că ar folosi degeaba resurse sistem.
- Utilizează câteva procese sau taskuri majoritatea resurselor? Pot fi acestea oprite? Le pot fi reduse prioritățile? Pot fi ele rafinate astfel încât să nu mai utilizeze așa multe resurse?

Depanarea performanței DB2 Connect

Dacă utilizatorii DB2 Connect se confruntă cu timpi mari de răspuns în cazul interogărilor voluminoase de pe serverele mainframe IBM, ar trebui examinate următoarele zone pentru cauze posibile ale problemelor de performanță:

1. Pentru interogările care au ca rezultat returnarea blocuri mari de date de la serverul mainframe IBM (uzual 32K de date și mai mult), asigurați-vă că parametrul de configurare manager bază de date RQRIOBLK este setat la 32767. Aceasta se poate face utilizând CLP (Command Line Processor), după cum urmează:

db2 update database manager configuration using RQRI0BLK 32767

2. Asigurați-vă că dimensiunea maximă RU definită în modul de definire IBMRDB este setată la o valoare potrivită. Se recomandă ca dimensiunea să nu fie mai mică de 4K pentru conexiunile ce folosesc hardware Token-ring. Pentru conexiunile ce folosesc hardware Ethernet, rețineți că dimensiunea maximă a cadrului Ethernet este 1536 octeți, ceea ce ar putea fi un factor limitator.

Ajustarea DB2 for z/OS

Puteți optimiza procesarea firelor de execuție inactive din z/OS. În V5, este permis până la 25,000 de clienți conectați concurrent. În toate aceste cazuri, numărul maxim de clienți care pot fi activi în același timp este însă 1999. Fiecare client stație de lucru poate rămâne conectat atunci când este inactiv; firul său de execuție este plasat într-un lanț inactiv la fiecare comitere.

Parametrii DSNZPARM CMTSTAT, CONDBAT și MAXDBAT influențează prelucrarea firelor de execuție. Pentru cea mai bună performanță, setați CMTSTAT pe INACTIVE, potriviți CONDBAT la numărul maxim de DBAT-uri conectate care furnizează o bună performanță și MAXDBAT la numărul maxim acceptabil de DBAT-uri active.

Creșterea ratelor de transfer de date DB2 Connect

În plus față de blocarea rândurilor pentru un set de rezultate interogare, DB2 for z/OS poate de asemenea returna mai multe astfel de blocuri de interogări în răspuns la o cerere OPEN sau FETCH pentru un client la distanță, cum ar fi DB2 Connect. În loc să trimită repetat clientul cereri la DB2 for z/OS care cer un bloc de date rând o dată, clientul poate cere opțional ca serverul să trimită înapoi un număr de blocuri de interogare în plus față de cel pe care-l trimite întotdeauna înapoi. Aceste blocuri de interogare sunt numite blocuri de interogare suplimentare.

Astfel, această nouă caracteristică permite clientului să reducă numărul de întoarceri pe linia de rețea, ceea ce reprezintă un cost important pentru performanțele rețelei. Scăderea numărului de cereri trimise de client către server pentru blocuri de interogare se traduce într-o mărire semnificativă a performanțelor. Această creștere de performanță se datorează faptului că comutarea între o linie de trimitere și de recepție este o operație costisitoare din punct de vedere al performanțelor. DB2 Connect poate acum exploata această îmbunătățire de performanță cerând blocuri extra de interogare DB2 for z/OS.

Pentru a profita din plin de întoarcerea blocurilor extra de interogare (fiecare putând având până la 32K octeți) pentru protocolul de rețea preferat de TCP/IP, extensiile de scalare fereastră au fost activate ca arhitectură sub RFC-1323 în DB2 Connect. Această caracteristică permite TCP/IP să ajusteze dinamic dimensiunile ferestrelor de trimitere și de recepție pentru a se adapta în mod eficient la eventualele cantități mari de date returnate pe calea blocurilor de interogare suplimentare.

Blocul de interogare suplimentar

Suport pentru interogare extra bloc pe servere cu DB2 for z/OS Versiunea 7 sau mai recentă este configurat prin parametrul EXTRA BLOCKS SRV în panoul de instalare DB2 DDF. Acest suport este configurat de calea de controlat cu numărul maxim de blocuri de interogări suplimentare pe care DB2 le poate trimite înapoi la un client pentru o cerere. Puteți seta acest parametru la o valoare între 0 și 100. Prin setarea parametrului la valoarea 0 se dezactivează returnarea de blocuri de interogare suplimentare. Valoarea implicită de 100 ar trebui utilizată întotdeauna pentru a avea cel mai mare avantaj al acestei caracteristici, exceptând unele comportamente specifice unei rețele care ar face ca setarea să nu mai fie ideală.

Pe partea de client, unde aplicația accesează DB2 for z/OS ori direct printr-o instalare co-localizată în DB2 Connect sau printr-o instalare server separată DB2 Connect, sunt mijloace variate pentru activarea suportul corespunzător DB2 Connect pe bază de cursor sau instrucțiune:

- Utilizarea unei dimensiuni a setului de rânduri interogare pentru un cursor.
- Utilizarea clauzei 'OPTIMIZE for N ROWS' în instrucțiunea select asociată cu un cursor
- Utilizarea clauzei 'FETCH FIRST N ROWS' în instrucțiunea select asociată cu un cursor

DB2 Connect poate activa suportul pentru rânduri suplimentare de interogare utilizând diferite API-uri SQL:

SQL încorporat

- Utilizatorul poate invoca suport bloc extra pentru o interogare specificând clauza 'OPTIMIZE for N ROWS' sau clauza 'FETCH FIRST N ROWS ONLY' sau ambele sau selectați însăși instrucțiunea.
- Cu clauza 'OPTIMIZE for N ROWS', DB2 for z/OS va încerca să pună în blocuri numărul dorit de rânduri de returnat la DB2 Connect, în funcție de setarea parametrului de instalare EXTRA BLOCKS SRV DDF. Aplicația poate alege să aducă mai mult de N rânduri deoarece DB2 for z/OS nu limitează numărul total de rânduri care ar putea fi returnat pentru rezultatul interogării setat la N.
- Clauza 'FETCH FIRST N ROWS ONLY' lucrează similar, cu excepția că setul de rezultate ale interogării este limitat la N rânduri de DB2 for z/OS. Aducerea a mai mult de N rânduri va avea ca rezultat codul SQL +100 (sfârșit date).

CLI/ODBC

- Utilizatorul poate invoca suportul pentru blocuri de interogare suplimentare, pentru o interogare, utilizând atributul său instrucțiune SQL_MAX_ROWS.
- Clauza 'FETCH FIRST N ROWS ONLY' este folosită în schimb pentru un server DB2 UDB pentru DB2 for z/OS 7.1 sau mai recent.
 - Pentru Versiunea 7, setul de rezultate ale interogării este limitat la N rânduri de DB2 pentru DB2 for z/OS. Încercarea de a aduce a mai mult de N rânduri va avea ca rezultat SQL_NO_DATA_FOUND.
 - Pentru Versiunea 8 sau mai recentă, CLI asigură că doar primele N rânduri sunt returnate aplicației prin managerul de cursor client.

JDBC Utilizatorul poate invoca suportul pentru blocuri de interogare suplimentare, pentru o interogare, utilizând metoda setMaxRows. Similar activării CLI/ODBC, DB2 Connect va eticheta clauza 'OPTIMIZE for N ROWS' pentru un server DB2 for z/OS 6.x. DB2 Connect va eticheta de asemenea clauza 'FETCH FIRST N ROWS ONLY' pentru un server DB2 for z/OS 7.1 sau mai recentă.

Scalarea ferestrei RFC-1323

Scalarea ferestrelor este suportată pe toate platformele Windows, Linux și UNIX care suportă extensiile RFC-1323 pentru TCP/IP. Puteți activa această caracteristică pe DB2 pentru Windows, Linux sau UNIX folosind variabila de registru DB2 DB2SORCVBUF. Pentru a activa scalarea ferestrelor, această variabilă de registru ar trebui setată la orice valoare peste 64K. De exemplu, pe DB2 pentru Windows, Linux, sau UNIX, puteți emite db2set DB2SORCVBUF =65537.

Dimensiunile maxime ale buffer-elor de trimitere și de primire sunt dependente de sistemul de operare. Pentru a se asigura că dimensiunile de buffer configurate au fost acceptate, utilizatorul poate să seteze parametrul de configurare al managerului de bază de date DIAGLEVEL la 4 (informativ) și să verifice mesajele din fișierul istoric al notificărilor de administrare.

Pentru ca scalarea ferestrelor să aibă efect trebuie activată la ambele capete ale unei conexiuni; sau și pe stația de lucru și gazdă, sau direct prin stiva TCP/IP a sistemului de operare, sau indirect prin produsul DB2. De exemplu, pentru DB2 for z/OS, scalarea ferestrelor poate fi activată prin sistemul de operare setând TCPRCVBUFRSIZE la orice valoare peste 64K. Dacă folosiți un IBM Data Server la distanță pentru a accesa o bază de date DB2 mainframe IBM printr-o stație de lucru a serverului DB2 Connect, puteți activa și scalarea ferestrei pe client. Cu același jeton, puteți activa scalarea ferestrei între un IBM Data Server la distanță și un server DB2 stație de lucru când nu este implicată nicio bază de date DB2 mainframe IBM.

În timp ce scalarea ferestrei este destinată să crească performanțele rețelei, este important de observat că îmbunătățirea așteptată de performanțe ale rețelei nu se materializează întotdeauna. Interacțiunea dintre factori cum ar fi dimensiunea cadrului folosit pentru adaptorul LAN Ethernet sau Token-ring, dimensiunea IP MTU și alte setări la rutere prin legătura de comunicație ar putea avea ca rezultat degradarea performanței o dată ce scalarea ferestrei a fost activată. De aceea, implicit, scalarea ferestrei este dezactivată, cu buffer-ele de recepție și de transmisie setate la 64K.

Trebuie să fiți pregătit să apreciați impactul pornirii scalării ferestrei și să realizați orice corecții necesare pentru rețea. Pentru o introducere despre ajustarea rețelei pentru performanțe îmbunătățite, vedeți <http://www.networking.ibm.com/>.

Conversia datelor gazdă

Când sunt transferate informații între diferite medii (cum ar fi Intel [Windows], IEEE [sistemele de operare Linux și UNIX], System z [VM, VSE, z/OS], IBM Power Systems [IBM i]), ar putea fi necesară convertirea tipurilor de date numerice (cum ar fi decimal, integer, floating point). Această conversie poate afecta performanța.

Costul CPU al conversiei datelor caracter pe un singur octet este în general mai mic decât acela al conversiei datelor numerice (când este necesară conversia datelor).

Costul conversiei de date pentru DATE/TIME/TIMESTAMP este aproape egal cu cel pentru CHAR pe un singur octet. Conversia datelor în virgulă mobilă (FLOATING) costă cel mai mult. Proiectantul de aplicație ar putea vrea să beneficieze de aceste aspecte la proiectarea unei aplicații pe baza DB2 Connect.

Dacă o tabelă de bază de date are o coloană definită 'FOR BIT DATA', datele de tip caracter transferate între aplicație și baza de date nu necesită nici o conversie de date. Aceasta poate fi folosită când arhivați date pe serverul bază de date mainframe IBM.

Tipurile de date pentru datele caracter

Datele caracter pot avea tipul de date CHAR sau VARCHAR. Care tip de date este mai eficient depinde de lungimea tipică de date din câmp:

- Dacă dimensiunea datelor propriu-zise variază semnificativ VARCHAR este mai eficient, deoarece CHAR adaugă caractere goale pentru a umple câmpul. Aceste caractere goale trebuie să fie transferate de-a lungul rețelei, la fel ca oricare alte caractere.
- Dacă dimensiunea datelor propriu-zise nu variază prea mult CHAR este mult mai eficient, deoarece fiecare câmp VARCHAR are câțiva octeți pentru lungime, care trebuie să fie transmiși.

Hardware-ul de rețea

Următoarele considerente se referă la hardware:

- Viteza rețelei sau mediul de transmisie

Performanțele se îmbunătățesc cu un mediu de transmisie mai rapid. De exemplu, următoarele valori sunt tipice pentru ratele de transfer brut al datelor:

Canal-la-canal (fibre optice)

4,0 MB/s

LAN 16 Mbps

2,0 MB/s

Canal-la-canal (obișnuit)

1,0 MB/s

LAN 4 Mbps

0,5 MB/s

Purtătoare T1 de mare viteză (1,544 Mbps)

0,193 MB/s

Linie telefonică rapidă la distanță 56 Kbps

0,007 MB/s

Model 19,6 Kbps

0,002 MB/s

Modem 9600 bps

0,001 MB/s

Rata transferului de date este limitată de cel mai încet mediu de transmisie din calea spre serverul bază de date mainframe IBM.

- Adaptorul de rețea sau controlorul de comunicații

Trebuie planificată cu grijă utilizarea memoriei adaptorului de rețea sau a controlerului de comunicații. În plus, trebuie să consultați un specialist în rețele pentru a vă asigura că controlerul are capacitatea de a manipula traficul suplimentar generat de DB2 Connect.

- Topologia rețelei

Dacă datele trec dintr-un LAN în alt LAN și dintr-o rețea în altă rețea, luați în considerare timpul pentru parcurgerea traseului. Punțile, ruterele și gateway-urile vor crește timpul de transfer. De exemplu, prin reducerea numărului punților traversate se reduce numărul de hopuri necesare pentru fiecare cerere.

Trebuie luată de asemenea în considerare distanța fizică dintre noduri. Chiar dacă un mesaj este transferat prin satelit, timpul de transfer este limitat de viteza lumini (3×10^8 m/s) și de distanța dus-întors dintre expeditor și destinatar.

- Traficul rețelei

Dacă lățimea de bandă a rețelei este complet utilizată, vor scădea atât timpul de răspuns, cât și rata transferului de date pentru o singură aplicație.

Pot apărea aglomerări în rețea atunci când datele se acumulează într-o anumită parte a rețelei; de exemplu, la un NCP vechi cu o dimensiune foarte mică a buffer-ului.

- Fiabilitatea rețelei

Dacă rata de erori a rețelei este mare, debitul rețelei va scădea și aceasta va duce la o performanță de nivel scăzut, din cauza retransmisiei.

Capitolul 21. Ajustarea performanței aplicației CLI/ODBC

CLI/ODBC este o interfață de programare a aplicațiilor SQL care poate fi apelată de aplicațiile dumneavoastră de bază de date. Funcțiile CLI invocă proceduri memorate DB2 care, în schimb, accesează tabelele de catalogare sistem.

Unele aplicații folosesc API-uri ODBC pentru a aduna informații de metadate care sunt folosite în procesarea ulterioară. Cele zece apeluri API de metadate care pot fi făcute sunt:

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns

Anumite aplicații CLI/ODBC care folosesc API-urile de metadate listate mai sus ar putea interoga toate obiectele din această bază de date. De exemplu, un apel `SQLTables` cere metadate pentru toate tabele din baza de date. Pe un sistem mare, asemenea cereri pot duce la mult trafic pe rețea, pot dura o perioadă considerabilă de timp și pot consuma o cantitate considerabilă din resursele serverului.

Pot fi folosite mai multe cuvinte cheie de inițializare CLI/ODBC pentru a limita cantitatea de date care vor fi returnate de apelurile inițiale API în timpul etapei "obținere informații" după conectarea la baza de date. Aceste cuvinte cheie pot fi setate de:

1. Editare manuală a fișierului `db2cli.ini`.
2. Prin schimbarea setărilor ODBC/CLI pentru baza de date folosind Configuration Assistant a clientului (pe acele platforme care le suportă).
3. Actualizând configurarea bazei de date CLI folosind linia de comandă a interfeței DBA.

Cuvintele cheie sunt:

- DBName
- TableType
- SchemaList
- SysSchema
- GrantorList
- GranteeList

Partea 5. Depanarea

Capitolul 22. Depanarea DB2 Connect

Mediul DB2 Connect implică mai multe produse software, hardware și de comunicație. Depanarea se realizează cel mai bine stabilind verdictul (locația erorii) printr-un proces de eliminare și rafinare a datelor disponibile.

După strângerea informațiilor relevante și bazându-vă pe selecția dumneavoastră de subiecte aplicabile, treceți la secțiunea de referință.

Strângerea informațiilor relevante

Depanarea include îngustarea domeniului problemei și investigarea cauzelor posibile. Un punct de start potrivit este adunarea informațiilor relevante și stabilirea a ceea ce se cunoaște, a datelor ce nu au fost strânse și ce căi de depanare puteți elimina. Răspundeți cel puțin la întrebările următoare.

- Conectările inițiale s-au realizat cu succes?
- Hardware-ul funcționează corespunzător?
- Căile de comunicație sunt operaționale?
- Au existat modificări ale rețelei de comunicație care ar fi putut face invalide intrările de director anterioare?
- A fost pornită baza de date?
- Este întreruptă comunicația este între unul sau mai mulți clienți și DB2 Connect Server (gateway); între gateway-ul DB2 Connect și serverul de bază de date mainframe IBM; sau între DB2 Connect Personal Edition și serverul de bază de date mainframe IBM.
- Ce puteți determina din conținutul mesajului și jetoanele returnate în mesaj?
- Folosirea uneltelor de diagnoză cum ar fi db2trc, db2pd sau db2support v-ar putea ajuta în acest moment?
- Alte mașini realizează taskuri similare care funcționează corect?
- Dacă acesta este un task la distanță, reușește executarea lui locală?

Conectarea inițială nu se realizează cu succes

Revedeți următoarele întrebări și asigurați-vă că pașii de instalare au fost urmați:

1. *Procesarea instalării s-a realizat cu succes?*
 - Au fost disponibile toate produsele software de cerințe preliminare?
 - A fost adecvat spațiul de memorie și de disc?
 - A fost instalat suportul pentru client la distanță?
 - A fost completată instalarea software-ului de comunicații fără nici o condiție de eroare?
2. *Pentru sistemele de operare UNIX, a fost creată o instanță a produsului?*
 - Ca root ați creat un utilizator și un grup care să devină proprietarul instanței și grupul sysadm?
3. *Dacă este cazul, informațiile de licență au fost procesate cu succes?*
 - Pentru sisteme de operare UNIX, ați editat fișierul nodelock și ați introdus parola pe care a livrat-o IBM?
4. *Configurațiile între serverul de bază de date mainframe IBM și stația de lucru au fost configurate corect?*
 - Există trei configurații care trebuie să fie luate în considerare:

- a. Configurația server bază de date mainframe IBM identifică solicitantul aplicației la server. Sistemul de gestiune server bază de date mainframe IBM va avea intrări catalog sistem care vor defini solicitantul în termeni de locație, protocol rețea și securitate.
 - b. Configurația stației de lucru DB2 Connect definește populația client pentru server și serverul mainframe IBM.
 - c. Configurarea stației de lucru client trebuie să aibă numele stației de lucru și protocolul de comunicație definite.
- Analiza problemelor pentru nerealizarea unei conexiuni inițiale include verificarea că numele PU (unitate fizică) sunt complete și corecte sau verificarea conexiunilor TCP/IP pentru care au fost specificate numărul corect de port și numele gazdei.
 - Administratorul server bază de date mainframe IBM și administratorul de rețea au utilitare disponibile pentru a diagnostica probleme.
5. *Aveți nivelul de autoritate cerut de sistemul de gestionare a bazei de date de pe serverul mainframe IBM pentru a folosi baza de date de pe mainframe IBM?*
 - Evaluați autoritatea de acces a utilizatorului, regulile pentru calificativele de tabele, rezultatele anticipate.
 6. *Dacă încercați să folosiți CLP (Command Line Processor) pentru a lansa instrucțiuni SQL pentru serverul de bază de date mainframe IBM, reușiți?*
 - Ați urmat procedura de a lega CLP la serverul bază de date mainframe IBM?

Au apărut probleme după conectarea inițială

Următoarele întrebări pot constitui un punct de plecare pentru a îngusta domeniul problemei.

1. *Există circumstanțe de operare speciale sau neobișnuite?*
 - Există o nouă aplicație?
 - Există noi proceduri utilizate?
 - Există modificări recente care pot afecta sistemul? De exemplu, au fost modificate produse sau aplicații software de la ultima rulare cu succes a aplicației sau scenariului?
 - Pentru programe de aplicație, ce interfață de aplicație de program (API) a fost folosită pentru a crea programul?
 - Au fost rulate alte aplicații care folosesc software-ul sau API-uri de comunicații pe sistemul utilizatorului?
 - A fost instalat recent un pachet de corecții? Dacă problema a apărut în timp ce un utilizator încerca folosirea unei caracteristici ce nu a mai fost folosită (sau încărcată) în sistemul de operare respectiv de când a fost instalată, stabiliți care este cel mai recent pachet de corecții IBM și încărcați-l *după* instalarea caracteristicii.
2. *Această eroare a mai apărut înainte?*
 - Există vreo rezolvare documentată pentru condiții de eroare anterioare?
 - Cine au fost participanții și dacă aceștia pot furniza informații de specialitate privind o posibilă evoluție a acțiunii?
3. *Ați încercat să folosiți comenzile software-ului de comunicații care returnează informații despre rețea?*
 - TCP/IP ar putea avea informații valoroase extrase folosind comenzi și demoni TCP/IP.
4. *Există informații returnate în SQLCA (SQL communication area) care vă pot fi utile?*
 - Problema care se ocupă de proceduri trebuie să includă pași de examinare a conținuturilor câmpurilor SQLCODE și SQLSTATE.
 - SQLSTATE permite programatorilor de aplicații să testeze pentru clase de erori care sunt comune pentru familia DB2 a produselor bază de date. Într-o rețea bază de date relațională distribuită acest câmp ar putea furniza o bază comună.

5. *A fost executat START DBM pe Server?* Adițional, asigurați-vă că mediul variabilei DB2COMM este setat corect pentru clienții care accesează serverul de la distanță.
6. *Există alte mașini care efectuează aceeași activitate și pot să se conecteze cu succes la server?* Numărul maxim de clienți care încearcă să se conecteze la server ar putea fi atins. Dacă se deconectează un client de la server poate după aceea să se conecteze clientul care mai înainte nu se putea conecta?
7. *Mașina are adresarea corespunzătoare?* Verificați dacă mașina este unică în rețea.
8. *Când se face conectarea de la distanță, este garantată autoritatea necesară clientului?* Conectarea la instanță ar putea avea succes, dar autorizarea ar fi putut să nu fie garantată la nivelul bază de date sau tabel.
9. *Este aceasta prima mașină de conectat la o bază de date la distanță?* În medii distribuite ruterele sau punțile dintre rețele ar putea să blocheze comunicațiile între client și server. De exemplu, la folosirea TCP/IP, asigurați-vă că puteți face PING la gazda la distanță.

Uneltele de diagnoză

Dacă întâlniți o problemă, puteți utiliza următoarele:

- Toate datele de diagnoză, cum ar fi fișierele dump, fișierele capcană, istoricele de eroare, fișierele de notificare și istoricele de alerte, se află în calea specificată de parametrul de configurare a managerului bazei de date pentru calea directorului datelor de diagnoză (**diagpath**):

Dacă valoarea acestui parametru de configurare este null, datele de diagnoză sunt scrise într-unul dintre următoarele directoare sau foldere:

- Pentru mediile Linux și UNIX: `INSTHOME/sqllib/db2dump`, unde `INSTHOME` este directorul home al instanței.
- Pentru mediile Windows suportate:
 - Dacă nu este setată variabila de mediu **DB2INSTPROF**, se folosește `x:\SQLLIB\DB2INSTANCE`, unde `x:\SQLLIB` este referința drive-ului și a directorului specificat în variabila de registru **DB2PATH**, iar valoarea **DB2INSTANCE** este numele instanței.

Notă: Directorul nu trebuie să se numească `SQLLIB`.

- Dacă nu este setată variabila de mediu **DB2INSTPROF**, se folosește `x:\DB2INSTPROF\DB2INSTANCE`, unde **DB2INSTPROF** este numele directorului cu profilul de instanță, iar **DB2INSTANCE** este numele instanței (implicit, valoarea **DB2INSTDEF** pe sistemele de operare Windows pe 32 de biți).
- Pentru sisteme de operare Windows, puteți folosi Event Viewer pentru a vizualiza istoricul de notificare administrare.
- Printre uneltele de diagnoză care pot fi folosite se numără **db2trc**, **db2pd**, **db2support** și **db2diag**.
- Pentru sisteme de operare Linux și UNIX, comanda **ps**, care returnează informații de stare proces despre procese active pentru ieșire standard.
- Pentru sisteme de operare UNIX, fișierul de bază care este creat în directorul curent când au loc erori severe. Acesta conține o imagine de memorie a procesului terminat și poate fi utilizat pentru a determina care funcții au cauzat eroarea.

Capitolul 23. Urmăririle DB2 în DB2 Connect

Acțiunile și operațiile de urmărire din mediul dumneavoastră de lucru vă pot oferi informații utile pentru depanarea unei probleme. Puteți obține, face dump și formata o urmărire realizată în produsul server de bază de date DB2. Facilitatea de urmărire este furnizată ca parte a produsului server de bază de date DB2.

Obținerea unei urme DB2 folosind db2trc

Comanda db2trc controlează facilitatea de urmărire furnizată cu DB2. Facilitatea de urmărire înregistrează informații despre operații și formatează aceste informații într-o formă ce poate fi citită.

Nu uitați că apare o regie suplimentară atunci când rulează o urmărire, așa că activarea facilității de urmărire poate afecta performanța sistemului.

În general, echipele IBM de dezvoltare și de suport software folosesc urmărirea DB2 pentru depanare. Puteți rula o urmărire pentru a obține informații referitoare la o problemă pe care o investigați, dar folosirea sa are rezultate limitate fără cunoașterea codului sursă DB2.

Cu toate acestea, este important să știți cum se pornește corect urmărirea și cum se face dump pentru fișierele de urmărire, în eventualitatea că vi se cere obținerea lor.

Notă: Va trebui să aveți autoritatea SYSADM, SYSCTRL sau SYSMAINT pentru a folosi db2trc

Pentru a vă face o idee care sunt opțiunile disponibile, executați comanda db2trc fără niciun parametru:

```
C:\>db2trc
Usage: db2trc (chg|clr|dmp|flw|fmt|inf|off|on) options
```

Pentru informații suplimentare despre un anumit parametru al comenzii db2trc, folosiți opțiunea -u. De exemplu, pentru a vedea informații suplimentare despre activarea urmăririi, executați următoarea comandă:

```
db2trc on -u
```

Aceasta va furniza informații despre toate opțiunile adiționale (etichetate ca "facilități") ce pot fi specificate la activarea urmăririi DB2.

Când se activează urmărirea, cea mai importantă opțiune este -L. Aceasta specifică dimensiunea buffer-ului de memorie care va fi folosit pentru a stoca informațiile urmărite. Dimensiunea buffer-ului poate fi specificată în octeți sau în megaocteți. (Pentru a specifica megaocteți, adăugați "M" sau "m" după valoare). Dimensiunea buffer-ului de urmărire trebuie să fie un număr de megaocteți egal cu o putere a lui doi. Dacă specificați o dimensiune care nu îndeplinește această cerință, dimensiunea buffer-ului va fi rotunjită automat în jos, la cea mai apropiată putere a lui doi.

Dacă buffer-ul este prea mic, se pot pierde informații. În mod implicit, sunt păstrate numai cele mai recente informații în cazul în care se umple buffer-ul. Dacă buffer-ul este prea mare, poate fi dificilă trimiterea fișierului la echipa IBM de suport software.

Dacă se urmărește o operație relativ scurtă (cum ar fi o conexiune la baza de date), de obicei este suficientă o dimensiune de aproximativ 8 MB:

```
C:\> db2trc on -l 8M
Trace is turned on
```

Dacă însă urmăriți o operație mai mare sau dacă se desfășoară o activitate intensă, poate fi necesar un buffer de urmărire mai mare.

Pe majoritatea platformelor, urmărirea poate fi activată în orice moment și funcționează așa cum s-a arătat mai sus. Există însă și anumite situații de care trebuie să țineți cont:

1. Pe sisteme cu mai multe partiții de bază de date, trebuie să rulați o urmărire pentru fiecare partiție fizică (nu logică) de bază de date.
2. Pe platformele HP-UX, Linux și Solaris, dacă este dezactivată urmărirea după ce a fost pornită instanța, la următoarea pornire a urmării va fi folosit un buffer foarte mic, indiferent de dimensiunea specificată. De exemplu, să spunem că ați pornit ieri urmărirea folosind comanda `db2trc on -l 8m`, ați colectat o urmărire și apoi ați dezactivat urmărirea (`db2trc off`). Astăzi doriți să rulați urmărirea cu buffer-ul de memorie setat la 32 megaocteți (`db2trc on -l 32m`) fără a opri și a reporni instanța. Veți constata că în acest caz urmărirea va avea un buffer mic. Pentru a rula eficient urmărirea pe aceste platforme, activați urmărirea după pornirea instanței cu dimensiunea de buffer de care aveți nevoie și “curățați” după aceea buffer-ul după cum este necesar.

Realizarea unui dump pentru un fișier de urmărire DB2

După ce a fost activată facilitatea de urmărire folosind opțiunea ON, toată activitatea ulterioară a instanței va fi urmărită.

În timp ce rulează urmărirea, puteți folosi opțiunea `clr` pentru a curăța buffer-ul de urmărire. Toate informațiile existente în buffer-ul de urmărire vor fi înlăturate.

```
C:\>db2trc clr
Trace has been cleared
```

După terminarea operației urmărite, folosiți opțiunea `dmp` urmată de un nume de fișier de urmărire pentru a realiza pe disc un dump al buffer-ului de memorie. De exemplu:

```
C:\>db2trc dmp trace.dmp
Trace has been dumped to file
```

Facilitatea de urmărire va continua să ruleze după ce a fost realizat dump-ul pentru buffer-ul de urmărire. Pentru a dezactiva urmărirea, folosiți opțiunea OFF:

```
C:\>db2trc off
Trace is turned off
```

Formatarea unui fișier de urmărire DB2

Fișierul de dump creat de comanda `db2trc dmp` este în format binar și nu poate fi citit. Pentru a verifica dacă un fișier de urmărire poate fi citit, formatați fișierul de urmărire binar pentru a afișa controlul fluxului și trimiteți ieșirea formatată la un dispozitiv null.

Exemplul următor prezintă comanda pentru realizarea acestui task:

```
db2trc flw example.trc nul
```

unde `example.trc` este un fișier binar care a fost produs folosind opțiunea `dmp`.

Ieșirea acestei comenzi vă va spune explicit dacă există vreo problemă privind citirea fișierului și dacă urmărirea a fost sau nu înfășurată.

În acest punct, fișierul de dump poate fi trimis la IBM Software Support. Acolo va fi formatat pe baza nivelului dumneavoastră de service pentru DB2. Însă uneori este posibil să vi se ceară să treceți fișierul de dump în format ASCII înainte de a-l trimite. Puteți face aceasta cu opțiunile flw și fmt. Trebuie să furnizați numele fișierului binar de dump și numele fișierului ASCII pe care doriți să-l creați:

```
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
```

Dacă această ieșire indică faptul că "Trace wrapped" este "YES", înseamnă că buffer-ul de urmărire nu a fost suficient de mare pentru a conține toate informațiile colectate în perioada de urmărire. În funcție de situație, o urmărire înfășurată poate fi suficient de bună. Dacă vă interesează cele mai recente informații (adică informațiile păstrate implicit dacă nu se specifică opțiunea -i), ceea ce se află în fișierul de urmărire poate fi suficient. Însă dacă vă interesează ceea ce s-a întâmplat la începutul perioadei de urmărire sau dacă vă interesează tot ce s-a întâmplat, poate fi necesar să executați din nou operația, cu un buffer de urmărire mai mare.

Puteți folosi anumite opțiuni când formatați un fișier binar într-un fișier text ce poate fi citit. De exemplu, puteți folosi `db2trc fmt -xml trace.dmp trace.fmt` pentru a converti datele binare și a trimite rezultatul într-un format parsabil XML. Celelalte opțiuni sunt prezentate detaliat în descrierea comenzii de urmărire (`db2trc`).

Alt lucru de care trebuie să țineți cont este faptul că în sistemele de operare Linux și UNIX DB2 va realiza automat pe disc un dump pentru buffer-ul de urmărire atunci când se oprește instanța din cauza unei erori grave. Astfel, dacă este activată urmărirea când o instanță se termină anormal, este creat un fișier în directorul de diagnoză și este denumit `db2trdmp.###`, unde `###` este numărul partiției de bază de date. Nu la fel se întâmplă pe platformele Windows. În aceste situații trebuie să creați manual dump-ul urmăririi.

Pentru a recapitula, următorul exemplu prezintă o secvență obișnuită de comenzi `db2trc`:

```
db2trc on -l 8M
db2trc clr
<Executați comenzile pentru re-crearea problemei>
db2trc dump db2trc.dmp
db2trc off
db2trc flw db2trc.dmp <nume_fișier>.flw
db2trc fmt db2trc.dmp <nume_fișier>.fmt
db2trc fmt -c db2trc.dmp <nume_fișier>.fmt
```

Capitolul 24. Fișierele de urmărire DRDA

Înainte de a analiza urmărirea DRDA, trebuie să înțelegi că DRDA este un standard deschis pentru definirea structurilor de date și de comunicație. De exemplu, DRDA cuprinde un set de reguli pentru organizarea datelor în vederea transmisiei și modul de comunicare a informațiilor care apar.

Aceste reguli sunt definite în următoarele manuale de referință:

- 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

Versiunile PDF ale acestor manuale sunt disponibile la www.opengroup.org.

Utilitarul **db2drdat** înregistrează datele schimbate între un DRDA Application Requestor (AR) și un DB2 DRDA Application Server (AS) (de exemplu între DB2 Connect și un server de bază de date gazdă sau server Power Systems).

Utilitarul de urmărire

Utilitarul **db2drdat** înregistrează datele schimbate între serverul DB2 Connect (în numele IBM Data Server) și serverul de bază de date mainframe IBM.

Ca un administrator de bază de date (sau dezvoltator de aplicații), ai putea considera util să înțelegi cum funcționează acest flux de date, deoarece aceste cunoștințe vă pot ajuta să determinați originea unei anumite probleme. Să presupunem că aflați în următoarea situație: lansați o instrucțiune de bază de date **CONNECT TO** pentru un server de bază de date mainframe IBM și comanda eșuează și primiți un cod de retur pentru eșec. Dacă înțelegi exact ce informații au fost trimise sistemului de gestiune server bază de date mainframe IBM, ai putea determina cauza eșecului chiar dacă informațiile codului retur sunt generale. Multe eșuări sunt cauzate de simple erori de utilizator.

Ieșirea de la **db2drdat** listează fluxurile de date schimbate între stația de lucru DB2 Connect și sistemul de gestionare a serverului de bază de date mainframe IBM. Datele trimise la serverul bază de date mainframe IBM sunt etichetate **SEND BUFFER** și datele primite de la un server bază de date mainframe IBM sunt etichetate **RECEIVE BUFFER**.

Dacă un buffer de primire conține informație **SQLCA**, va fi urmat de o interpretare formatată a acestei date și etichetată **SQLCA**. Câmpul **SQLCODE** al unui **SQLCA** este valoarea *nemapată* returnată de serverul de bază de date mainframe IBM. Buffer-ele de trimitere și de primire sunt aranjate de la cel mai vechi la cel mai recent în fișier. Fiecare buffer are:

- ID-ul de proces
- O etichetă **SEND BUFFER**, **RECEIVE BUFFER** sau **SQLCA**. Prima comandă **DDM** sau obiect dintr-un buffer are eticheta **DSS TYPE**.

Datele care rămân în buffer-ele de trimitere sau de primire sunt divizate în 5 coloane care conțin:

- Un număr de octeți.
- Coloanele 2 și 3 reprezintă fluxurile de date DRDA schimbate între cele două sisteme, în ASCII sau EBCDIC.
- O reprezentare ASCII a coloanelor 2 și 3.

- O reprezentare EBCDIC a coloanelor 2 și 3.

ieșirea urmăririi

Utilitarul db2drdat scrie următoarele informații în *fișierul de urmărire*:

- -r
 - Tip de răspuns/obiect DRDA
 - Buffer de primire
- -s
 - Tip de cerere DRDA
 - Buffer de trimitere
- -c
 - SQLCA
- Informație de eroare TCP/IP
 - Cod de retur funcție de primire
 - Gravitate
 - Protocol folosit
 - API folosit
 - Funcție
 - Număr eroare

Notă:

1. O valoare zero pentru codul de ieșire indică faptul că s-a terminat cu succes comanda, iar o valoare nenulă indică faptul că nu este așa.
2. Câmpurile returnate variază în funcție de API-ul folosit.
3. Câmpurile returnate variază în funcție de platforma pe care rulează DB2 Connect, chiar și pentru același API.
4. În cazul în care comanda db2drdat trimite ieșirea către fișierul care deja există, vechiul fișier va fi șters, exceptând situația în care permisiunile fișierului împiedică ștergerea lui.

Analizarea fișierului cu ieșirea urmăririi

Într-o urmărire db2drdat sunt capturate următoarele informații:

- ID-ul procesului (PID) al aplicației client
- RDB_NAME catalogat în directorul de servicii de conexiuni la baze de date (DCS).
- CCSID-urile DB2 Connect
- CCSID-urile serverului de bază de date mainframe IBM
- Sistemul de gestionare a serverului bazei de date mainframe IBM cu care comunică sistemul DB2 Connect.

Primul buffer conține comenzile Exchange Server Attributes (EXCSAT) și Access RDB (ACCRDB) trimise la sistemul de gestiune server bază de date mainframe IBM. Trimite aceste comenzi ca rezultat al comenzii de bază de date CONNECT TO. Următorul buffer conține răspunsul pe care l-a primit DB2 Connect de la sistemul de gestionare a serverului de bază de date mainframe IBM. Conține EXCSATRD (Exchange Server Attributes Reply Data) și un ACCRDBRM (Access RDB Reply Message).

EXCSAT

Comanda EXCSAT conține numele stației de lucru a clientului specificat de obiectul SRVNAM (Server Name), care este punctul de cod X'116D', conform specificației

DDM. Comanda EXCSAT se află în primul buffer. În comanda EXCSAT, valorile X'9481A292' (codate în CCSID 500) sunt traduse în *mască* după ce s-a înlăturat X'116D'.

Comanda EXCSAT conține de asemenea obiectul EXTNAM (External Name), care este deseori plasat în informațiile de diagnoză de pe sistemul de gestionare a bazei de date mainframe IBM. Este alcătuit dintr-un ID de aplicație pe 20 de octeți, urmat de un ID de proces de 8 octeți (sau un ID de proces pe 4 octeți și un ID de fir de execuție pe 4 octeți). Este reprezentat de punctul de cod X'115E', în acest exemplu valoarea sa fiind completată de db2bp cu spații, urmate de 000C50CC. Pe un Linux sau UNIX IBM Data Server, această valoare poate fi corelată cu comanda ps, care returnează informațiile de stare despre procese despre procesele active la ieșirea standard.

ACCRDB

Comanda ACCRDB conține RDB_NAME din obiectul RDBNAM, care este punctul de cod X'2110'. Comanda ACCRDB urmează comenzii EXCSAT din primul buffer. În comanda ACCRDB, valorile X'E2E3D3C5C3F1' sunt traduse în STLEC1 după ce s-a înlăturat X'2110'. Aceasta corespunde câmpului de nume al bazei de date destinație din directorul DCS.

Șirul de contabilizare are punctul de cod X'2104'.

Setul de coduri configurat pentru stația de lucru DB2 Connect este afișat prin localizarea obiectului CCSID CCSIDSBC (CCSID pentru caractere pe un octet) cu punctul de cod X'119C' în comanda ACCRDB. În acest exemplu, CCSIDSBC este X'0333', ceea ce înseamnă 819.

În comanda ACCRDB sunt prezente de asemenea obiectele CCSIDDBC (CCSID pentru caractere pe doi octeți) și CCSIDMBC (CCSID pentru caractere pe octeți amestecați) adiționale, cu punctele de cod X'119D' și X'119E'. În acest exemplu, CCSIDSBC este X'04B0', ceea ce înseamnă 1200, iar CCSIDMBC este X'0333', ceea ce înseamnă 819.

EXCSATRD și ACCRDBRM

Valorile CCSID sunt de asemenea returnate de la serverul bază de date mainframe IBM în Access RDB Reply Message (ACCRDBRM) în al doilea buffer. Acest buffer conține EXCSATRD urmat de ACCRDBRM. Fișierul de ieșire exemplu conține două valori CCSID pentru sistemul server bază de date mainframe IBM. Valorile sunt 1208 (pentru caractere pe un octet și octet mixt) 1200 (pentru caractere pe doi octeți).

Dacă DB2 Connect nu recunoaște pagina de cod ce vine înapoi de la serverul de bază de date mainframe IBM, utilizatorul primește SQLCODE -332, împreună cu paginile de cod sursă și țintă. Dacă serverul de bază de date mainframe IBM nu recunoaște setul de coduri trimis de DB2 Connect, returnează VALNSPRM (Parameter Value Not Supported, cu punctul de cod DDM X'1252'), care se traduce în SQLCODE -332 pentru utilizator.

ACCRDBRM conține de asemenea parametrul PRDID (Product-specific Identifier, cu punctul de cod X'112E'). Valoarea este X'C4E2D5F0F8F0F1F5', ceea ce înseamnă DSN08015 în EBCDIC. Conform standardelor, DNS este DB2 for z/OS. Este indicat de asemenea numărul de versiune. ARI este DB2 Server for VSE & VM, SQL este baza de date DB2 sau DB2 Connect, iar QSQ este DB2 for IBM i.

Exemple de fișier cu ieșire a urmării

Figurile următoare prezintă o ieșire exemplu ce ilustrează câteva schimburi de fluxuri de date DRDA între stațiile de lucru DB2 Connect și un server de bază de date gazdă sau System i. Din punctul de vedere al utilizatorului, a fost emisă o comandă bază de date `CONNECT TO` folosind procesorul liniei de comandă (CLP).

Figura 13 la pagina 143 folosește DB2 Connect Enterprise Edition Versiunea 9.1 și DB2 for z/OS Versiunea 8 peste o conexiune 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	40404040404040	@@@@@	

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

Figura 13. Exemplu de ieşire de urmărire (conexiune 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.....
0010	00162110E2E3D3C5	C3F1404040404040	..!.....@@@
0020	40404040404000C	11A1D9858799F485	@@@.....
0030	A599000A11A09585	A6A39695vr....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!.\$.
0010	00172135C7F9F1C1	F0C4F3C14BD7C1F8	..!5.....K...
0020	F806030221064600	162110E2E3D3C5C3	...!.F.!.....
0030	F140404040404040	4040404040000C11	.@@@@@@@@@... 1
0040	2EE2D8D3F0F9F0F0	F000D002FD8E3C4/... .SQL09000....QTD
0050	E2D8D3C1E2C30016	00350006119C03335.....3 SQLASC.....
0060	0006119D04B00006	119E0333003C21043.

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..
0010	000D002FD8E3C4E2	D8D3F3F7F0000C11	../. ..QTDSQL370...
0020	2EC4E2D5F0F8F0F1	F5001600350006115... .DSN08015.....
0030	9C04B80006119E04	B80006119D04B000
0040	0C11A0D5C5E6E3D6	D540400006212524@...!%\$...NEWTON
0050	34001E244E000624	4C00010014244D00	4..\$N..\$L...\$M.+...<.....(.
0060	06244FFFFF000A11	E8091E768301BE00	.\$0.....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

Figura 14. Exemplu de ieşire de urmărire (conexiune TCP/IP) - continuare


```

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 @@@@AAAAAHLV.... .....<.....
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                sl.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

```

Figura 15. Exemplu de ieşire de urmărire (conexiune TCP/IP) - continuare

```
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 4F544C2000FFFFFFE	2704DSNXOTL+!.<.....
0020	0C00000000000000 00FFFFFFF000000
0030	0000000000572020 2057202020202020W 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\$..	..}.....

Figura 16. Exemplu de ieşire de urmărire (conexiune TCP/IP) - continuare

Informațiile ulterioare de buffer pentru urmărirea DRDA

Puteți analiza buffer-urile de trimitere și de primire rezultate pentru informații adiționale. Următoarea cerere conține o comitere. Comanda commit determină sistemul de gestionare a serverului de bază de date mainframe IBM să comită unitatea curentă de lucru. Al patrulea buffer este primit de la sistemul de gestiune bază de date server bază de date mainframe IBM ca rezultat al unei comiteri sau derulări înapoi. Conține mesajul de răspuns sfârșit unitate de lucru (ENDUOWRM), care indică faptul că unitatea curentă de lucru s-a încheiat.

În acest exemplu, intrarea de urmărire 12 conține un SQLCA null, indicat de punctul de cod DDM X'2408' urmat de X'FF'. Un SQLCA null (X'2408FF') indică succesul (SQLCODE 0).

Figura 13 la pagina 143 arată un exemplu a unui buffer primit care conține un SQLCA de eroare la intrarea de urmărire 16.

Partea 6. Mesajele

Capitolul 25. Probleme obișnuite DB2 Connect

Acest subiect listează cele mai comune simptome de probleme de conexiune întâlnite la folosirea DB2 Connect. În fiecare caz, vi se oferă:

- O combinație dintre un număr mesaj și un cod retur (sau un cod retur specific protocolului) asociat mesajului. Fiecare combinație de mesaj și cod retur are un antet separat și anteturile sunt ordonate după numărul mesajului și apoi după codul retur.
- Un simptom, de obicei sub formă listării unui mesaj exemplu.
- O soluție sugerată, indicând cauza probabilă a erorii. În unele cazuri, ar putea fi livrată mai de o soluție sugerată.

SQL0965 sau SQL0969

Simptom

Pot fi lansate mesajele SQL0965 și SQL0969 cu diferite coduri de retur din DB2 for IBM i, DB2 for z/OS și DB2 Server for VM and VSE.

Când întâlniți oricare din mesaje, ar trebui să examinați codul SQL original din documentația referitoare la produsul server de bază de date care emite mesajul.

Soluție Codul SQL primit de la serverul bază de date mainframe IBM nu poate fi translatat. Corectați problema, pe baza codului erorii, apoi relansați comanda care a eșuat.

SQL5043N

Simptom

Suportul pentru unul sau mai multe protocoale de comunicație a eșuat să pornească. Totuși funcția managerului bazei de date nucleu a fost pornită cu succes.

Poate protocolul TCP/IP nu este pornit pe serverul DB2 Connect. Se poate să fi existat o conexiune cu succes client anterioară.

Dacă `diaglevel = 4`, atunci fișierele istoric `db2diag` ar putea conține o intrare similară, de exemplu:

```
2001-05-30-14.09.55.321092 Instance:svtdbm5 Node:000
PID:10296(db2tcpm) Appid:none
common_communication sqlcctcpconnmgr_child Probe:46
DIA3205E Socket address "30090" configured in the TCP/IP
services file and
required by the TCP/IP server support is being used by another
process.
```

Soluție Acest avertisment este un simptom care semnalează că DB2 Connect, care acționează ca server pentru clienți la distanță, are probleme în manevrarea unuia sau mai multor comunicații client. Aceste protocoale pot fi TCP/IP și altele, mesajul indicând de obicei că nu este configurat corect unul dintre protocoalele de comunicație definite pentru DB2 Connect.

Deseori cauza poate fi că variabila de profil DB2COMM nu este definită sau este definită incorect. În general, problema este cauzată de nepotrivirea dintre variabila DB2COMM și numele definite în configurația managerului bazei de date (de exemplu, `svcname` sau `nname`).

Un scenariu posibil este să aveți o conexiune anterioară reușită și apoi să primiți mesajul de eroare SQL5043, fără ca nimic să se fi schimbat în configurație. Aceasta ar putea apărea folosind protocolul TCP/IP, când sistemul la distanță termină anormal conexiunea dintr-un motiv oarecare. Când aceasta se întâmplă, este posibil

să pară că încă există o conexiune pe client și că ar putea fi restaurată fără intervenții suplimentare emițând comenzile arătate mai jos.

Mai probabil, unul din clienții care se conectează la serverul DB2 Connect încă are un mâner pe portul TCP/IP. Pe fiecare mașină care este conectată la serverul DB2 Connect, introduceți următoarele comenzi:

```
db2 terminate
db2stop
```

SQL30020

Simptom

SQL30020N Execution failed because of a Distributed Protocol Error that will affect the successful execution of subsequent commands and SQL statements.

Soluții Ar trebui contactat service-ul pentru această eroare. Rulați comanda db2support înainte de a contacta service-ul.

SQL30060

Simptom

SQL30060N "<authorization-ID>" does not have the privilege to perform operation "<operation>".

Soluție La conectarea la DB2 for z/OS, tabelele CDB nu au fost actualizate corect.

SQL30061

Simptom

Conectarea la o locație greșită de server bază de date mainframe IBM - nu poate fi găsită nicio bază de date destinație.

Soluție Un nume de server de bază de date eronat ar putea fi specificat în intrarea directorului DCS. Când se întâmplă aceasta, este returnat SQLCODE -30061 către aplicație.

Verificați nodul, baza de date și intrările director DCS DB2. Câmpul nume bază de date destinație din intrarea directorului DCS trebuie să corespundă numelui bazei de date bazat pe platformă. De exemplu, pentru o bază de date DB2 for z/OS, numele care urmează să fie folosit trebuie să fie cel folosit în câmpul BSDS (Boot Strap Data Set) "LOCATION=locname", care este de asemenea furnizat în mesajul DSNL004I (LOCATION=location) când este pornit DDF (Distributed Data Facility).

Comenzile corecte pentru un nod TCP/IP sunt:

```
db2 catalog tcpip node <nume_nod> remote <nume_sau_adresă_gazdă>
server <nt_port_sau_nume_serviciu>
db2 catalog dcs database <nume_local> as <nume_real_bd>
db2 catalog database <nume_local> as <alias> at <nume_node>
authentication server
```

Pentru a vă conecta la baza de date lansați apoi:

```
db2 connect to <alias> user <nume_utilizator> using <parolă>
```

SQL30081N cu Cod retur 79

Simptom

```
SQL30081N A communication error has been detected.
Communication protocol
being used: "TCP/IP". Communication API being used: "SOCKETS".
Location
```



```
where the error was detected: ". Communication function
detecting the error:
"connect". Protocol specific error code(s): "79", "*", "*".
SQLSTATE=08001
```

Soluție (soluții)

Această eroare poate avea loc în cazul unui client la distanță care eșuează la conectarea la un server DB2 Connect. De asemenea, poate apărea la conectarea de la serverul DB2 Connect la un server de bază de date mainframe IBM.

1. Variabila de profil DB2COMM ar putea fi setată incorect pe serverul DB2 Connect. Verificați aceasta. De exemplu, comanda `db2set db2comm=tcPIP` ar trebui să apară în `sqliib/db2profile` la rularea DB2 Enterprise Server Edition pe AIX.
2. Ar putea fi o nepotrivire între numele de serviciu TCP/IP și specificațiile de număr de port la IBM Data Server și serverul DB2 Connect. Verificați intrările în fișierele `services` TCP/IP pe amândouă mașinile.
3. Verificați că DB2 este pornit pe serverul DB2 Connect. Setati `diaglevel` al configurației managerului de bază de date la valoarea 4, folosind comanda:
`db2 update dbm cfg using diaglevel 4`

După oprirea și repornirea DB2, uitați-vă în fișierul istoric `db2diag` pentru a verifica dacă au fost pornite comunicațiile TCP/IP DB2. Ar trebui să vedeți o ieșire similară cu:

```
2001-02-03-12.41.04.861119 Instance:svtdbm2 Node:00
PID:86496(db2sysc) Appid:none
common_communication sqlcctcp_start_listen Probe:80
DIA3000I "TCPIP" protocol support was successfully started.
```

SQL30081N cu codul de eroare specific protocolului 10032

Simptom

```
SQL30081N A communication error has been detected.
Communication protocol
being used: "TCP/IP". Communication API being used: "SOCKETS".
Location
where the error was detected: "9.21.85.159". Communication
function detecting
the error: "send". Protocol specific error code(s): "10032",
"*, "*".
SQLSTATE=08001
```

Soluție Acest mesaj de eroare ar putea fi primit la încercarea de deconectarea de la o mașină un comunicațiile TCP/IP deja au eșuat. Corecți problema cu subsistemele TCP/IP. pe majoritatea mașinilor, repornirea protocolului TCP/IP este modul de rezolvare a problemei. Ocazional, ar putea fi necesară reciclarea întregii mașini.

SQL30082 RC=24 în timpul CONNECT

Simptom

SQLCODE -30082 Nu este corect numele de utilizator sau parola furnizată.

Soluție Asigurați-vă că este furnizată parola corectă în instrucțiunea `CONNECT`, dacă este necesar. Parola nu este disponibilă pentru trimiterea la baza de date de pe serverul destinație. Trebuie să fie trimisă o parolă de la IBM Data Server la baza de date de pe serverul destinație. Pe anumite platforme, de exemplu AIX, parola poate fi obținută doar dacă este furnizată în instrucțiunea `CONNECT`.

Partea 7. Anexe

Anexa A. Privire generală asupra informațiilor tehnice DB2

Informațiile tehnice DB2 sunt disponibile prin următoarele unelte și metode:

- Centrul de informare DB2
 - Subiecte (subiecte de task, de concept și de referință)
 - Ajutor pentru uneltele DB2
 - Programe exemplu
 - Îndrumările
- Cărțile DB2
 - Fișiere PDF (descărcabile)
 - Fișiere PDF (de pe DVD-ul DB2 PDF)
 - Cărți tipărite
- Ajutor pentru linia de comandă
 - Ajutor pentru comandă
 - Ajutor pentru mesaj

Notă: Subiectele din Centrul de informare DB2 sunt actualizate mai frecvent decât cărțile tipărite sau în format PDF. Pentru a beneficia de cele mai recente informații, instalați actualizările de documentație pe măsură ce devin disponibile sau consultați Centrul de informare DB2 de la ibm.com.

Puteți găsi de asemenea informații tehnice DB2 suplimentare, cum ar fi note tehnice, documente White Paper și publicații IBM Redbooks, la ibm.com. Vizitați situl cu biblioteca software-ului DB2 Information Management, la <http://www.ibm.com/software/data/sw-library/>.

Reacția dumneavoastră cu privire la documentație

Apreciem reacția dumneavoastră privind documentația DB2. Dacă aveți sugestii privind îmbunătățirea documentației DB2, trimiteți un e-mail la db2docs@ca.ibm.com. Echipa pentru documentația DB2 citește toate reacțiile, dar nu vă poate răspunde direct. Pentru o mai bună înțelegere a ceea ce vreți să ne comunicați, furnizați exemple specifice. Dacă trimiteți o reacție referitoare la un anumit subiect sau fișier de ajutor, includeți titlul subiectului și URL-ul.

Nu folosiți această adresă de e-mail pentru a contacta DB2 Customer Support. Dacă aveți o problemă tehnică referitoare la DB2 care nu este tratată în documentație, contactați centrul de service IBM local pentru asistență.

Biblioteca tehnică DB2 în format tipărit sau PDF

Tabelele următoare prezintă biblioteca DB2 disponibilă în IBM Publications Center, la www.ibm.com/shop/publications/order. Manualele DB2 Versiunea 9.7 în engleză sau traduse în format PDF pot fi descărcate de la www.ibm.com/support/docview.wss?rs=71&uid=swg2700947.

Deși în tabele sunt identificate cărțile tipărite disponibile, cărțile pot să nu fie disponibile în țara sau regiunea dumneavoastră.

Numărul de formular este mărit de fiecare dată când este actualizat un manual. Aveți grijă să citiți versiunea cea mai recentă a manualelor, conform listei de mai jos.

Notă: *Centrul de informare DB2* este actualizat mai frecvent decât cărțile tipărite sau în format PDF.

Tabela 20. Informații tehnice DB2

Nume	Număr format	Disponibil tipărit	Ultima actualizare
<i>Administrative API Reference</i>	SC27-2435-01	Da	Noiembrie 2009
<i>Administrative Routines and Views</i>	SC27-2436-01	Nu	Noiembrie 2009
<i>Call Level Interface Guide and Reference, Volume 1</i>	SC27-2437-01	Da	Noiembrie 2009
<i>Call Level Interface Guide and Reference, Volume 2</i>	SC27-2438-01	Da	Noiembrie 2009
<i>Command Reference</i>	SC27-2439-01	Da	Noiembrie 2009
<i>Data Movement Utilities Guide and Reference</i>	SC27-2440-00	Da	August, 2009
<i>Data Recovery and High Availability Guide and Reference</i>	SC27-2441-01	Da	Noiembrie 2009
<i>Database Administration Concepts and Configuration Reference</i>	SC27-2442-01	Da	Noiembrie 2009
<i>Database Monitoring Guide and Reference</i>	SC27-2458-01	Da	August, 2009
<i>Database Security Guide</i>	SC27-2443-01	Da	Noiembrie 2009
<i>DB2 Text Search Guide</i>	SC27-2459-01	Da	Noiembrie 2009
<i>Developing ADO.NET and OLE DB Applications</i>	SC27-2444-01	Da	August, 2009
<i>Developing Embedded SQL Applications</i>	SC27-2445-01	Da	Noiembrie 2009
<i>Developing Java Applications</i>	SC27-2446-01	Da	Noiembrie 2009
<i>Developing Perl, PHP, Python, and Ruby on Rails Applications</i>	SC27-2447-00	Nu	August, 2009
<i>Developing User-defined Routines (SQL and External)</i>	SC27-2448-01	Da	Noiembrie 2009
<i>Getting Started with Database Application Development</i>	GI11-9410-01	Da	Noiembrie 2009
<i>Getting Started with DB2 Installation and Administration on Linux and Windows</i>	GI11-9411-00	Da	August, 2009
<i>Globalization Guide</i>	SC27-2449-00	Da	August, 2009

Tabela 20. Informații tehnice DB2 (continuare)

Nume	Număr format	Disponibil tipărit	Ultima actualizare
<i>Instalarea serverelor DB2</i>	GC27-2455-01	Da	Noiembrie 2009
<i>Instalarea clienților IBM Data Server</i>	GC27-2454-00	Nu	August, 2009
<i>Referințe mesaje, volumul 1</i>	SC27-2450-00	Nu	Noiembrie 2009
<i>Referințe mesaje, volumul 2</i>	SC27-2451-00	Nu	Noiembrie 2009
<i>Net Search Extender Administration and User's Guide</i>	SC27-2469-01	Nu	Noiembrie 2009
<i>Partitioning and Clustering Guide</i>	SC27-2453-01	Da	Noiembrie 2009
<i>pureXML Guide</i>	SC27-2465-01	Da	Noiembrie 2009
<i>Query Patroller Administration and User's Guide</i>	SC27-2467-00	Nu	August, 2009
<i>Spatial Extender and Geodetic Data Management Feature User's Guide and Reference</i>	SC27-2468-00	Nu	August, 2009
<i>SQL Procedural Languages: Application Enablement and Support</i>	SC27-2470-01	Da	August, 2009
<i>SQL Reference, Volume 1</i>	SC27-2456-01	Da	Noiembrie 2009
<i>SQL Reference, Volume 2</i>	SC27-2457-01	Da	Noiembrie 2009
<i>Troubleshooting and Tuning Database Performance</i>	SC27-2461-01	Da	Noiembrie 2009
<i>Modernizarea la DB2 Versiunea 9.7</i>	SC27-2452-01	Da	Noiembrie 2009
<i>Visual Explain Tutorial</i>	SC27-2462-00	Nu	August, 2009
<i>Ce este nou pentru Versiunea 9.7</i>	SA22-1406-01	Da	Noiembrie 2009
<i>Workload Manager Guide and Reference</i>	SC27-2464-01	Da	August, 2009
<i>XQuery Reference</i>	SC27-2466-01	Nu	Noiembrie 2009

Tabela 21. Informații tehnice privind DB2 Connect

Nume	Număr format	Disponibil tipărit	Ultima actualizare
<i>Instalarea și configurarea DB2 Connect Personal Edition</i>	SA22-1404-01	Da	Noiembrie 2009
<i>Instalarea și configurarea serverelor DB2 Connect</i>	SA22-1405-01	Da	Noiembrie 2009
<i>Ghidul utilizatorului DB2 Connect</i>	SA22-1403-01	Da	Noiembrie 2009

Tabela 22. Informații tehnice Information Integration

Nume	Număr format	Disponibil tipărit	Ultima actualizare
<i>Information Integration: Administration Guide for Federated Systems</i>	SC19-1020-02	Da	August, 2009
<i>Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i>	SC19-1018-04	Da	August, 2009
<i>Information Integration: Configuration Guide for Federated Data Sources</i>	SC19-1034-02	Nu	August, 2009
<i>Information Integration: SQL Replication Guide and Reference</i>	SC19-1030-02	Da	August, 2009
<i>Information Integration: Introduction to Replication and Event Publishing</i>	GC19-1028-02	Da	August, 2009

Comandarea cărților DB2 tipărite

Dacă aveți nevoie de cărți DB2 tipărite, le puteți cumpăra online în multe, dar nu în toate țările sau regiunile. Puteți oricând comanda cărți tipărite DB2 de la reprezentantul local IBM. Nu uitați că anumite cărți de pe DVD-ul *DB2 PDF Documentation* nu sunt disponibile în varianta tipărită. De exemplu, nici unul dintre volumele *Referințe mesaje DB2* nu este disponibil ca o carte tipărită.

Versiunile tipărite ale multora dintre cărțile DB2 de pe DVD-ul *DB2 PDF Documentation* pot fi comandate contra cost de la IBM. În funcție de locul de unde plasați comanda, puteți comanda cărți online, de la IBM Publications Center. Dacă în țara sau regiunea dumneavoastră nu este disponibilă comandarea online, puteți oricând să comandați cărți tipărite DB2 de la reprezentantul local IBM. Rețineți că nu toate cărțile de pe DVD-ul *DB2 PDF Documentation* sunt disponibile în varianta tipărită.

Notă: Cea mai recentă și mai cuprinzătoare documentație DB2 se află în Centrul de informare DB2 de la <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7>.

Comandarea cărților tipărite DB2:

- Pentru a afla dacă puteți să comandați cărți tipărite DB2 online în țară sau regiune, verificați centrul de publicații IBM la <http://www.ibm.com/shop/publications/order>. Trebuie să selectați o țară, regiune sau limbă pentru a accesa informațiile de comandare publicații și apoi să urmați instrucțiunile pentru locația dumneavoastră.
- Pentru a comanda cărți tipărite DB2 de la reprezentantul local IBM:
 1. Localizați informațiile de contact pentru reprezentantul local pe unul dintre următoarele surse web:
 - Directorul IBM cu contactele din întreaga lume la www.ibm.com/planetwide
 - Situl web al publicațiilor IBM la <http://www.ibm.com/shop/publications/order>. Veți avea nevoie să selectați țara, regiunea sau limba pentru a accesa în mod corespunzător paginile de bază (home) ale publicațiilor pentru locația dumneavoastră. Din această pagină, urmați legătura "About this site".
 2. Când sunați, specificați că doriți să comandați publicația DB2.

3. Furnizați reprezentantului titlurile și numerele de formular ale cărților pe care doriți să le comandați. Pentru titluri și numere de formular, vedeți “Biblioteca tehnică DB2 în format tipărit sau PDF” la pagina 157.

Afișarea ajutorului pentru starea SQL din linia de comandă a procesorului

Produsele DB2 întorc o valoare SQLSTATE pentru condiții care ar putea fi rezultatul unei instrucțiuni SQL. Ajutorul pentru SQLSTATE explică semnificația stărilor SQL și a codurilor de clase de stări SQL.

pentru a porni ajutorul pentru o stare SQL, deschideți procesorul linie de comandă și introduceți:

```
? sqlstate sau ? cod clasă
```

unde *sqlstate* reprezintă o stare SQL validă de 5 cifre și *cod clasă* reprezintă primele două cifre ale stării SQL.

De exemplu, ? 08003 afișează ajutorul pentru starea SQL 08003, iar ? 08 afișează ajutorul pentru codul clasei 08 .

Accesarea diferitelor versiuni de Centru de informare DB2

Pentru subiectele DB2 Versiunea 9.7, URL-ul *Centrului de informare DB2* este <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/>.

Pentru subiectele DB2 Versiunea 9.5, URL-ul *Centrului de informare DB2* este <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>.

Pentru subiectele DB2 Versiunea 9.1, URL-ul *Centrului de informare DB2* este <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>.

Pentru subiectele DB2 Versiunea 8, mergeți la URL-ul *Centrului de informare DB2* la: <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>.

Afișarea subiectelor în limba preferată în Centrul de informare DB2

Centrul de informare DB2 încearcă să afișeze subiecte în limba specificată în preferințele browser-ului. Dacă un subiect nu a fost tradus în limba preferată, centrul de informare DB2 afișează subiectul în engleză.

- Pentru a afișa subiectele în limba preferată în browser-ul Internet Explorer:
 1. În Internet Explorer, faceți clic pe butonul **Tools** —> **Internet Options** —> **Languages...** Se deschide fereastra Language Preferences.
 2. Asigurați-vă că limba preferată este specificată în prima intrare din lista de limbi.
 - Pentru a adăuga o nouă limbă, faceți clic pe butonul **Add...**

Notă: Adăugarea unei limbi nu garantează că pe calculator se află fonturile necesare pentru afișarea subiectelor în limba preferată.

 - Pentru a muta o limbă în vârful listei, selectați o limbă și faceți clic pe butonul **Move Up** până când limba este prima în lista de limbi.
 - 3. Reîmprospătați pagina pentru a afișa Centrul de informare DB2 în limba dumneavoastră preferată.
- Pentru a afișa subiectele în limba preferată într-un browser Firefox sau Mozilla:

1. Selectați butonul din secțiunea **Languages** în dialogul **Tools** —> **Options** —> **Advanced**. Este afișat panoul Languages în fereastra Preferences.
2. Asigurați-vă că limba preferată este specificată în prima intrare din lista de limbi.
 - Pentru a adăuga o nouă limbă la listă, faceți clic pe butonul **Add...** pentru a selecta o limbă din fereastra Add Languages.
 - Pentru a muta o limbă în vârful listei, selectați o limbă și faceți clic pe butonul **Move Up** până când limba este prima în lista de limbi.
3. Reîmprospătați pagina pentru a afișa Centrul de informare DB2 în limba dumneavoastră preferată.

Pe unele browser-e și combinații de sistem de operare, trebuie de asemenea să modificați setările regionale ale sistemului de operare la Locale-ul și limba dorită.

Actualizarea Centrului de informare DB2 instalat pe calculatorul dumneavoastră sau pe un server din intranet

Un Centru de informare DB2 instalat local trebuie actualizat periodic.

Înainte de a începe

Un Centru de informare DB2 Versiunea 9.7 trebuie să fie deja instalat. Pentru detalii vedeți subiectul “Instalarea centrului de informare DB2 folosind vrăjitorul DB2 Setup” în *Instalarea serverelor DB2*. Toate cerințele preliminare și cerințele care se aplică instalării Centrului de informare se aplică și pentru actualizarea Centrului de informare.

Despre acest task

Un Centru de informare DB2 existent poate fi actualizat automat sau manual:

- Actualizările automate - actualizează caracteristicile și limbile Centrului de informare existente. Un avantaj suplimentar al actualizărilor automate este că Centrul de informare este indisponibil pentru o perioadă de timp minimă în timpul actualizării. În plus, actualizările automate pot fi setate să ruleze ca parte a altor joburi batch care rulează periodic.
- Actualizările manuale - trebuie folosite când vreți să adăugați caracteristici sau limbi în timpul procesului de actualizare. De exemplu, un Centru de informare local a fost instalat inițial cu ambele limbi, franceză și engleză, iar acum doriți să instalați limba germană; o instalare manuală va instala germana, și va actualiza caracteristicile și limbile Centrului de informare existente. Dar, o actualizare manuală necesită să opriți manual centrul de informare, să-l actualizați și să-l reporniți. Centrul de informare nu este disponibil pe întreg procesul de actualizare.

Procedura

Acest subiect detaliază procesul pentru actualizările automate. Pentru actualizarea manuală, vedeți subiectul “Actualizarea manuală a centrului de informare DB2 instalat pe calculatorul dumneavoastră sau pe un server intranet”.

Pentru a actualiza automat Centrul de informare DB2 instalat pe calculatorul dumneavoastră sau pe un server din intranet:

1. Pe sistemele de operare Linux:
 - a. Navigați în calea în care este instalat Centrul de informare. În mod implicit, Centrul de informare DB2 este instalat în directorul `/opt/ibm/db2ic/V9.7`.
 - b. Navigați din directorul de instalare în directorul `doc/bin`.

- c. Rulați scriptul ic-update:
ic-update
2. Pe sisteme de operare Windows:
 - a. Deschideți o fereastră de comandă.
 - b. Navigați în calea în care este instalat Centrul de informare. În mod implicit, Centrul de informare DB2 este instalat în directorul <Program Files>\IBM\DB2 Information Center\Version 9.7, unde <Program Files> este locația directorului Program Files.
 - c. Navigați din directorul de instalare în directorul doc\bin.
 - d. Rulați fișierul ic-update.bat:
ic-update.bat

Rezultate

Centrul de informare DB2 repornește automat. Dacă sunt actualizări disponibile, Centrul de informare afișează subiectele noi și actualizate. Dacă nu sunt disponibile actualizări pentru centru de informare, se adaugă un mesaj în istoric. Fișierul istoric se află în directorul doc\eclipse\configuration. Numele fișierului istoric este un număr generat aleatoriu. De exemplu, 1239053440785.log.

Actualizarea manuală a Centrului de informare DB2 instalat pe calculatorul dumneavoastră sau pe un server din intranet

Dacă ați instalat Centrul de informare DB2 local, puteți să descărcați și să instalați actualizările de la IBM.

Despre acest task

Pentru a actualiza manual *Centrul de informare DB2* instalat local, trebuie să:

1. Opriți *Centrul de informare DB2* pe calculatorul dumneavoastră și apoi porniți-l în modul independent. Când rulați Centrul de informare în modul independent, nu poate fi accesat de alți utilizatori din rețea, astfel că puteți să aplicați actualizările. Versiunea Workstation (stație de lucru) a Centrului de informare DB2 rulează întotdeauna în modul independent.
2. Folosiți caracteristica Actualizare pentru a vedea ce actualizări sunt disponibile. Dacă există actualizări pe care trebuie să le instalați, puteți folosi caracteristica Actualizare pentru a le obține și instala.

Notă: Dacă mediul dumneavoastră necesită instalarea actualizărilor *Centrului de informare DB2* pe o mașină care nu este conectată la internet, oglindiți situl de actualizare într-un sistem de fișiere local folosind o mașină care este conectată la internet și pe care este instalat *Centrul de informare DB2*. De asemenea, dacă actualizările de documentație vor fi instalate de mulți utilizatori din rețeaua dumneavoastră, puteți reduce timpul necesar pentru o actualizare individuală oglindind local situl de actualizare și creând un proxy pentru situl de actualizare.

Dacă sunt disponibile pachete de actualizare, folosiți caracteristica Actualizare pentru a obține pachetele respective. Însă caracteristica Actualizare este disponibilă numai în modul independent.

3. Opriți Centrul de informare independent și reporniți pe calculator *Centrul de informare DB2*.

Notă: Pe Windows 2008, Windows Vista (și mai noi), comenzile listate mai târziu în această secțiune, trebuie rulate ca administrator. Pentru a deschide un prompt de comandă sau o unealtă grafică având privilegiile complete de administrator, faceți clic dreapta pe scurtătură și apoi selectați **Run as administrator**.

Procedura

Pentru a actualiza *Centrul de informare DB2* instalat pe calculatorul dumneavoastră sau pe un server din intranet:

1. Opriți *Centrul de informare DB2*.
 - În Windows, faceți clic pe **Start** → **Control Panel** → **Administrative Tools** → **Services**. Apoi faceți clic dreapta pe serviciul **Centrul de informare DB2** și selectați **Stop**.
 - În Linux, introduceți următoarea comandă:
`/etc/init.d/db2icdv97 stop`
2. Porniți Centrul de informare în modul independent.
 - În Windows:
 - a. Deschideți o fereastră de comandă.
 - b. Navigați în calea în care este instalat Centrul de informare. În mod implicit, *Centrul de informare DB2* este instalat în directorul `Program_Files\IBM\DB2 Information Center\Version 9.7`, unde `Program_Files` reprezintă locația directorului Program Files.
 - c. Navigați din directorul de instalare în directorul `doc\bin`.
 - d. Rulați fișierul `help_start.bat`:
`help_start.bat`
 - În Linux:
 - a. Navigați în calea în care este instalat Centrul de informare. În mod implicit, *Centrul de informare DB2* este instalat în directorul `/opt/ibm/db2ic/V9.7`.
 - b. Navigați din directorul de instalare în directorul `doc/bin`.
 - c. Rulați scriptul `help_start`:
`help_start`

Se deschide browser-ul Web implicit al sistemului pentru a afișa Centrul de informare independent.
3. Faceți clic pe butonul **Actualizare** (🔄). (JavaScript™ trebuie activat în browser.) În panoul din partea dreaptă a Centrului de informare, faceți clic pe **Găsire actualizări**. Este afișată o listă de actualizări pentru documentația existentă.
4. Pentru a iniția procesul de instalare, bifați selecțiile pe care doriți să le instalați și apoi faceți clic pe **Instalare actualizări**.
5. După terminarea procesului de instalare, faceți clic pe **Sfârșit**.
6. Opriți Centrul de informare independent:
 - În Windows, navigați la directorul de instalare `doc\bin` și rulați fișierul `help_end.bat`:
`help_end.bat`

Notă: Fișierul batch `help_end` conține comenzile necesare pentru oprirea în siguranță a proceselor pornite cu fișierul batch `help_start`. Nu folosiți Ctrl-C sau altă metodă pentru a opri `help_start.bat`.

 - În Linux, navigați la directorul `doc/bin` din directorul de instalare și rulați scriptul `help_end`:
`help_end`

Notă: Scriptul `help_end` conține comenzile necesare pentru oprirea în siguranță a proceselor pornite cu scriptul `help_start`. Nu folosiți altă metodă pentru a opri scriptul `help_start`.

7. Reporniți *Centrul de informare DB2*.

- În Windows, faceți clic pe **Start** → **Control Panel** → **Administrative Tools** → **Services**. Apoi faceți clic dreapta pe serviciul **Centrul de informare DB2** și selectați **Start**.
- În Linux, introduceți următoarea comandă:
`/etc/init.d/db2icdv97 start`

Rezultate

Este afișat *Centrul de informare DB2* cu subiectele noi și actualizate.

Îndrumarele DB2

Îndrumarele DB2 vă ajută să vedeți diverse aspecte ale produselor DB2. Lecțiile oferă instrucțiuni pas cu pas.

Înainte de a începe

Puteți vizualiza versiunea XHTML a îndrumarului din centrul de informare la <http://publib.boulder.ibm.com/infocenter/db2help/>.

Unele lecții folosesc date sau cod de exemplu. Vedeți îndrumarul pentru o descriere a cerințelor preliminare pentru operațiile sale specifice.

Îndrumarele DB2

Pentru a vizualiza îndrumarul, faceți clic pe titlu.

“pureXML” în *pureXML Guide*

Setați o bază de date DB2 astfel încât să stocați date XML și să realizați operații de bază cu depozitul de date XML native.

“Visual Explain” în *Visual Explain Tutorial*

Analizați, optimizați și ajustați instrucțiunile SQL pentru performanțe mai bune folosind Visual Explain.

Informații privind depanarea DB2

Este disponibilă o gamă largă de informații privind depanarea și determinarea problemelor, pentru a vă ajuta la folosirea produselor bază de date DB2.

Documentația DB2

Pentru informații privind depanarea, vedeți *DB2 Troubleshooting Guide* sau secțiunea Elementele de bază ale bazei de date, din Centrul de informare *DB2*. Acolo găsiți informații referitoare la izolarea și identificarea problemelor folosind utilitarele și uneltele de diagnosticare DB2, soluții pentru unele dintre cele mai comune probleme și alte sfaturi pentru rezolvarea problemelor care pot apărea atunci când folosiți produsele bază de date DB2.

Situl Web DB2 Technical Support

Consultați situl Web DB2 Technical Support dacă aveți probleme și doriți ajutor la găsirea cauzei și a soluțiilor posibile. Situl de suport tehnic are legături la cele mai noi publicații DB2, documente TechNote, APAR-uri, pachete de corecții și alte

resurse. Puteți căuta prin această bază de date de cunoștințe pentru a găsi soluțiile posibile pentru problemele dumneavoastră.

Accesați situl Web DB2 Technical Support, la http://www.ibm.com/software/data/db2/support/db2_9/

Termenii și condițiile

Permisunile pentru utilizarea acestor publicații sunt acordate în conformitate cu următorii termeni și condiții.

Utilizare personală: Puteți reproduce aceste publicații pentru utilizarea personală, necomercială, cu condiția ca toate anunțurile de proprietate să fie păstrate. Nu puteți să distribuiți, să afișați sau să realizați lucrări derivate din aceste Publicații sau dintr-o porțiune a lor fără consimțământul explicit primit de la IBM.

Utilizare comercială: Puteți reproduce, distribui și afișa aceste publicații doar în cadrul întreprinderii dumneavoastră, cu condiția ca toate anunțurile de proprietate să fie păstrate. Nu puteți să realizați lucrări derivate din aceste Publicații, nici să reproduceți, să distribuiți sau să afișați aceste Publicații sau o porțiune a lor în afara întreprinderii dumneavoastră fără consimțământul explicit primit de la IBM.

În afara celor acordate expres prin această permisiune, nu se acordă nicio altă permisiune, licență sau drept, explicite sau implicite, pentru aceste Publicații sau orice informații, date, software sau alte elemente pe care le conțin și care reprezintă o proprietate intelectuală.

IBM își rezervă dreptul de a retrage permisiunea acordată aici oricând consideră că folosirea Publicațiilor este în detrimentul intereselor sale sau când IBM constată că instrucțiunile de mai sus nu au fost respectate.

Nu puteți descărca, exporta sau reexporta aceste informații decât în deplină conformitate cu legile și regulamentele aplicabile, inclusiv toate legile și regulamentele de export ale Statelor Unite.

IBM NU OFERĂ NICIO GARANȚIE PRIVIND CONȚINUTUL ACESTOR PUBLICAȚII. PUBLICAȚIILE SUNT FURNIZATE "CA ATARE", FĂRĂ NICIUN FEL DE GARANȚIE, EXPLICITĂ SAU IMPLICITĂ, INCLUZÂND, DAR FĂRĂ A SE LIMITA LA ELE, GARANȚIILE IMPLICITE DE VANDABILITATE, DE NEÎNCĂLCARE A UNOR DREPTURI SAU NORME ȘI DE POTRIVIRE PENTRU UN ANUMIT SCOP.

Anexa B. Observații

Aceste informații au fost elaborate pentru produse și servicii oferite în S.U.A. Informațiile despre produsele IBM se bazează pe informațiile disponibile la momentul publicării pentru prima dată a acestui document și se pot modifica.

Este posibil ca IBM să nu ofere în alte țări produsele, serviciile sau caracteristicile discutate în acest document. Luați legătura cu reprezentantul IBM local pentru informații despre produsele și serviciile disponibile în zona dumneavoastră. Referirea la un produs, program sau serviciu IBM nu înseamnă că se afirmă sau că se sugerează faptul că poate fi folosit numai acel produs, program sau serviciu IBM. Poate fi folosit în loc orice produs, program sau serviciu care este echivalent din punct de vedere funcțional și care nu încalcă dreptul de proprietate intelectuală al IBM. Însă evaluarea și verificarea modului în care funcționează un produs, program sau serviciu non-IBM ține de responsabilitatea utilizatorului.

IBM poate avea brevete sau aplicații în curs de brevetare care să acopere subiectele descrise în acest document. Faptul că vi se furnizează acest document nu înseamnă că vi se acordă licența pentru aceste brevete. Puteți trimite întrebări referitoare la licență, în scris, la:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Pentru întrebări privind licența pentru informațiile pe doi octeți (DBCS), contactați departamentul IBM de proprietate intelectuală din țara dumneavoastră sau trimiteți întrebări în scris la:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
3-2-12, Roppongi, Minato-ku, Tokyo 106-8711 Japan

Paragraful următor nu se aplică în cazul Marii Britanii sau al oricărei alte țări/regiuni în care aceste prevederi sunt incompatibile cu legile locale: INTERNATIONAL BUSINESS MACHINES CORPORATION OFERĂ ACEASTĂ PUBLICAȚIE “CA ATARE”, FĂRĂ NICIUN FEL DE GARANȚIE, EXPLICITĂ SAU IMPLICITĂ, INCLUZÂND, DAR FĂRĂ A SE LIMITA LA ELE, GARANȚIILE IMPLICITE PRIVIND NEÎNCĂLCAREA UNUI DREPT, VANDABILITATEA SAU POTRIVIREA PENTRU UN ANUMIT SCOP. Unele state nu permit declinarea răspunderii pentru garanțiile explicite sau implicite în anumite tranzacții; de aceea, este posibil ca această declarație să nu fie valabilă în cazul dumneavoastră.

Aceste informații pot include inexactități tehnice sau erori tipografice. Se efectuează modificări periodice la informațiile incluse aici; aceste modificări vor fi încorporate în noi ediții ale publicației. IBM poate aduce îmbunătățiri și/sau modificări produsului (produselor) la care se referă această publicație în orice moment, fără notificare.

Orice referire în cadrul acestor informații la alte situri Web decât cele IBM este oferită doar pentru comoditate și nu reprezintă în nici un fel susținerea acelor situri Web. Materialele de pe siturile Web respective nu fac parte din materialele pentru acest produs IBM, iar utilizarea acestor situri Web se face pe propriul risc.

IBM poate folosi sau distribui informațiile pe care le furnizați în orice mod crede că este corespunzător, fără ca aceasta să implice vreo obligație pentru dumneavoastră.

Deținătorii de licență pentru acest program care doresc să obțină informații despre el pentru a permite: (i) schimbul de informații între programe create independent și alte programe (inclusiv cel de față) și (ii) folosirea reciprocă a informațiilor schimbate trebuie să contacteze:

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

Aceste informații pot fi disponibile, cu respectarea termenilor și condițiilor, inclusiv, în unele cazuri, cu plata unei taxe.

Programul licențiat descris în acest document și toate materialele licențiate disponibile pentru el sunt furnizate de IBM conform termenilor din IBM Customer Agreement, IBM International Program License Agreement sau orice acord echivalent încheiat între noi.

Toate datele de performanță din acest document au fost determinate într-un mediu controlat. De aceea, rezultatele obținute în alte medii de funcționare pot fi diferite. Anumite măsurători s-ar putea să fi fost făcute pe sisteme în faza de dezvoltare și nu există nici o garanție că aceste măsurători vor fi la fel pe sistemele în faza finală. Mai mult, unele măsurători pot fi estimări obținute prin extrapolare. Rezultatele reale pot varia. Utilizatorii acestui document trebuie să verifice datele aplicabile pentru mediul lor specific.

Informațiile referitoare la produsele non-IBM au fost obținute de la furnizorii produselor respective, din anunțurile lor publicate sau din alte surse disponibile public. IBM nu a testat aceste produse și nu poate confirma nivelul performanței, compatibilitatea sau alte calități pretinse ale acestor produse non-IBM. Întrebările despre capacitățile produselor non-IBM trebuie să fie adresate furnizorilor acelor produse.

Toate declarațiile privind direcția viitoare sau intențiile IBM pot fi schimbate sau retractate fără notificare, reprezentând doar scopuri și obiective.

Aceste informații pot conține exemple de date și rapoarte folosite în operațiile comerciale de zi cu zi. Pentru a fi cât mai complete, exemplele includ nume de persoane, de companii, de mărci și de produse. Toate aceste nume sunt fictive și orice asemănare cu nume sau adrese folosite de o întreprindere reală este pură coincidență.

LICENȚĂ COPYRIGHT:

Aceste informații conțin exemple de programe de aplicație în limbaj sursă, care ilustrează tehnici de programare pentru diverse platforme de operare. Puteți copia, modifica și distribui aceste exemple de programe după cum doriți, fără vreo plată către IBM, dacă o faceți pentru dezvoltarea, folosirea, în scop de marketing sau de distribuire a programelor de aplicație în conformitate cu interfața de programare a aplicațiilor pentru platforma de operare pentru care au fost scrise exemplele de program. Aceste exemple nu au fost testate amănunțit în toate condițiile. Ca urmare, IBM nu poate garanta sau sugera fiabilitatea, capacitatea de service sau funcționalitatea acestor programe. Programele exemplu sunt furnizate "AȘA CUM SUNT", fără niciun fel de garanție. IBM nu va putea fi făcut responsabil pentru niciun fel de pagube ce ar putea avea ca și cauză folosirea de către dumneavoastră a programelor exemplu.

Fiecare copie sau porțiune din aceste programe eșantion sau lucrările derivate din ele trebuie să conțină un anunț de copyright, după cum urmează:

© (numele companiei dumneavoastră) (an). Părți din acest cod sunt derivate din IBM Corp. Sample Programs. © Copyright IBM Corp. *_introduceți anul sau anii_*. Toate drepturile sunt rezervate.

Mărci comerciale

IBM, sigla IBM și *ibm.com* sunt mărci comerciale sau mărci comerciale înregistrate deținute de International Business Machines Corp în multe jurisdicții din toată lumea. Alte nume de servicii și produse pot fi mărci comerciale deținute de IBM sau alte companii. O listă curentă de mărci comerciale IBM este disponibilă pe Web la Copyright and trademark information la www.ibm.com/legal/copytrade.shtml.

Următorii termeni sunt mărci comerciale sau mărci comerciale înregistrate ale altor companii

- Linux este o marcă comercială înregistrată deținută de Linus Torvalds în Statele Unite, în alte țări sau ambele.
- Java și toate mărcile comerciale și emblemele bazate pe Java sunt mărci comerciale deținute de Sun Microsystems, Inc. în Statele Unite, în alte țări sau ambele.
- UNIX este o marcă comercială înregistrată deținută de Open Group în Statele Unite și în alte țări.
- Intel, sigla Intel, Intel Inside[®], sigla Intel Inside, Intel[®] Centrino[®], sigla Intel Centrino, Celeron[®], Intel[®] Xeon[®], Intel SpeedStep[®], Itanium[®] și Pentium[®] sunt mărci comerciale sau mărci comerciale înregistrate deținute de Intel Corporation în Statele Unite, în alte țări sau ambele.
- Microsoft, Windows, Windows NT[®] și logo-ul Windows sunt mărci comerciale deținute de Microsoft Corporation în Statele Unite, în alte țări sau ambele.

Alte nume de companii, produse sau servicii pot fi mărci comerciale sau mărci de servicii ale altora.

Index

Caractere speciale

&&

fișier de mapare SQLCODE 57

A

actualizare multilocație

Activare 51

Control Center 52

manager punct sync 52

testarea 52

unitate de lucru distribuită (DUOW) 51

actualizări

Centrul de informare DB2 162, 163

directoare bază de date 27

afinități client

configurarea 114

ajustare

baze de date gazdă 120

DB2 Connect 83

DB2 for z/OS 123

parametrii

agentpri 119

dir_cache 119

maxagents 119

MAXDARI 119

numdb 119

rqrioblk 119

rețele 121

ajutor

configurare limbă 161

instrucțiuni SQL 161

alertele de sănătate

DB2 pentru z/OS 68

alias bază de date client 64

aplicații

compus SQL 91

legare 47

performanța

proiectare aplicație 91

proceduri memorate 91

proiectare 91

Web

DB2 Connect 15

aplicații client

recuperarea comunicațiilor 79

aplicații web

DB2 Connect 15

proceduri memorate 18

Arhitectura reprezentării datelor tip caracter (CDRA) 7

ATOMIC compus SQL

nesuportat în DB2 Connect 91

autentificare 33

DB2 Connect 44

declarație REVOKE 44

director bază de date sistem 27

privire generală 41

tipuri

CLIENT 41, 43

DATA_ENCRYPT 41

autentificare (*continuare*)

tipuri (*continuare*)

KERBEROS 41

SERVER 41

SERVER_ENCRYPT 41

SERVER_ENCRYPT_AES 41

valori implicite 41

validare 41

autoritate BINDADD

DB2 Connect 47

autoritate CREATE IN COLLECTION NULLID 47

autorizare

legare 47

B

baze de date

ajustare 120

aliasuri

director bază de date sistem 27

foaie de lucru pentru personalizarea directorului 33

gazdă 4

grupare cereri 91

nume

director bază de date sistem 27

director DCS 29

foaie de lucru pentru personalizarea directorului 33

obiect RDBNAM 140

unelte de performanță 85

baze de date destinație

nume 29, 33

baze de date federalizate

cereri distribuite 10

baze de date gazdă

accesare folosind DB2 Connect Personal Edition 11

conectivitate

disponibilitate înaltă 77

încărcare echilibrare 77

blocare

date 91

blocuri de interogare suplimentare

parametru EXTRA BLOCKS SRV 123

privire generală 123

blocuri de interogări

creșterea ratei de transfer a datelor DB2 Connect 123

buffer de trimitere

date urmărire 139

C

cărți

comandare 160

CCSID (identificator set de caractere codat)

suport bidirecțional

detalii 29

Centrul de informare DB2

actualizare 162, 163

limbi 161

versiuni 161

cereri bază de date
 grupare pentru performanță 91

cereri distribuite
 privire generală 10

clauza FOR FETCH ONLY
 instrucțiune SELECT 91

clienți
 echilibrare încărcări de lucru 107
 rerutare automată client
 conexiuni la DB2 for z/OS 110
 suport Sysplex
 limitări 117

clienți server de date IBM
 echilibrare încărcare de lucru 107
 rerutare automată client
 pe partea client 110

cod de eroare SQL0969 151
 cod de eroare SQL30020 151
 cod de eroare SQL30061 151
 cod de eroare SQL30081N 151
 cod de eroare SQL30082 151
 cod eroare SQL0965 151
 cod eroare SQL30060 151
 cod eroare SQL30073 151
 cod eroare SQL5043N 151

comanda ACCRDB 140
 comanda ACCRDBRM 140
 comanda ACCSEC 140
 comanda atributelor server exchange 140
 comanda db2drdat
 Fișier de ieșire 139

comanda db2trc
 formatare ieșire urmărire 136
 privire generală 135
 realizarea unui dump pentru ieșirea urmăririi 136

comanda de comitere 140

comanda de sistem START MVS 68
 comanda de sistem STOP MVS 68

comanda EXCSAT 140
 comanda EXCSATRD 140
 comanda FORCE 64
 comanda GET SNAPSHOT
 privire generală 62

comanda LIST DCS APPLICATIONS
 ieșire 64

comanda SECCHK 140
 comandarea cărților DB2 160

comandă ps
 obiect EXTNAM 140
 privire generală 133

comeni
 ACCRDBRM 140
 EXCSATRD 140

comenzi
 ACCRDB 140
 ACCSEC 140
 comitere 140
 db2drdat
 privire generală 139
 db2trc
 formatare fișier urmărire 136
 obținere urmă 135
 EXCSAT 140
 GET SNAPSHOT
 privire generală 62
 SECCHK 140

comitere în două faze
 Activare 51
 port resynch folosit de conexiunile TCP/IP 28

comunicații
 recuperare 79

concentrator de conexiuni
 agenți de lucru 97
 comparație punere conexiuni în pool 101
 DB2 Connect 101
 gestionarea conexiunilor 95
 privire generală 95, 97

conexiuni
 DB2 Connect Enterprise Edition 14
 direct la gazde 11
 direct la mainframe IBM 11
 eșecuri
 rerutare client automat 81
 găzduire directă System z 12
 gestionare 95
 IBM i direct 12
 pool
 avantaje 97
 concentratori de conexiuni 97
 privire generală 95

restabilire
 DB2 Connect Enterprise Edition 14
 direct la gazdă 11

conexiuni de încredere
 CLI/ODBC 38
 comutarea utilizatorilor prin CLI/ODBC 39
 DB2 Connect 37

configurație
 conexiuni gazdă 11
 modificări parolă 43

conflict
 resurse de sistem 122

contexte de încredere
 suport CLI/ODBC 38
 suport DB2 Connect 37

Control Center
 actualizare multilocație 52

conversie
 gazdă 125

cotare de nivel
 performanța 85

CPU-uri
 unelte de performanță 85

cuvânt cheie CURRENTPACKAGESET CLI/ODBC 43

D

date
 blocare 91
 flux
 DB2 Connect 7, 85
 suport fus orar 29
 surse 10
 transferare
 între gazde și stații de lucru 55
 performanța 126
 rate 85, 126

DB2 Connect
 concentratori de conexiuni 101
 configurare
 IBM Power Systems 53
 mainframe IBM 53
 System z 53

- DB2 Connect *(continuare)*
 - Enterprise Edition
 - aplicații web 15
 - manageri de tranzacție compatibilă XA 53
 - Monitorizări de procesare tranzacție 21
 - server de aplicații Java 17
 - servere de conectivitate 14
 - servere web 18
 - îmbunătățiri
 - funcții 3
 - mutarea datelor 55
 - privire generală 3
 - produse 3
 - scenarii server de conectivitate 11
 - securitate 37
 - suport gazdă 11
 - suport mainframe 11
 - suport Sysplex 103, 107
 - suport System i 11
 - utilitare de administrare 5
- DB2 for z/OS
 - monitorul de sănătate
 - acțiuni recomandate 70
 - obiecte de alertă 74
 - oprire 69
 - pornire 69
 - privire generală 68
 - reîmprospătare 69
 - sumare alerte 72
 - opțiunea DYNAMICRULES (BIND) 43
 - securitate 43
 - valori director nod 28
- debit
 - tranzacții 85
- depanare
 - conexiuni 131, 132
 - DB2 Connect 131, 151
 - informații online 165
 - îndrumare 165
 - performanța 122
 - strângere informații 131
 - urmăriri
 - DRDA 142, 146
 - obținere folosind comanda db2trc 135
- despre această carte v
- determinarea problemei
 - conexiune 131
 - informații disponibile 165
 - îndrumare 165
 - post-conexiune 132
 - unelte de diagnoză
 - privire generală 133
- dezvoltarea aplicațiilor
 - IBM Data Server Driver Package 11
 - ODBC 11
 - proiectare aplicație 91
- dimensiune bloc
 - DB2 Connect 119
- dimensiune bloc pagină 119
- directoare
 - bază de date a sistemului
 - actualizare 27
 - valori 27
 - personalizare 33
- directoare bază de date
 - actualizare 27
 - DCS (Database Connection Services) 27

- directoare bază de date *(continuare)*
 - intrări multiple 33
 - nod 27
- director bază de date sistem
 - actualizare 27
 - valori 27
- Director DCS (Database Connection Services)
 - vedeți Director DCS (servicii conectare baze de date) 29
- Director DCS (Servicii Conectare la Baza de date)
 - actualizare intrări 27
 - valori 29
- disponibilitate înaltă
 - DB2 Connect 75
- documentație
 - fișiere PDF 157
 - privire generală 157
 - termeni și condiții de utilizare 166
 - tipărite 157
- DRDA (Distributed Relational Database Architecture)
 - acces de date 7
 - DB2 Connect 7
 - privire generală 7
- driver-e IBM Data Server
 - echilibrare încărcare de lucru 107
 - rerutare automată client
 - pe partea client 110
- DSS (Decision Support System) 139
- dump urmă la un fișier
 - privire generală 136

E

- echilibrare încărcare de lucru
 - configurare clienți 108
 - pe partea client 107
- element monitorizare nume aplicație 64
- erori
 - depanare 131
- exemple
 - concentratori de conexiuni 97
 - concentratori XA 97

F

- facilitate de control al accesului la resurse (RACF)
 - autentificare 44
- FDOCA (Formatted Data Object Content Architecture) 7
- fișă de lucru
 - personalizare director 33
- fișier dcs1ari.map 57
- fișier dcs1dsn.map 57
- fișier dcs1qsq.map 57
- fișier ddc400.lst 47
- fișier ddcsmvs.lst 47
- fișier ddcsvm.lst 47
- fișier ddcsvse.lst 47
- fișier de configurare db2dsdriver 108
- fișiere de bază
 - determinarea problemei 133
- fus orar
 - privire generală 29

G

- gâtuiuri
 - performanța 85

gături (*continuare*)
tranzacții 85
Gestionare date distribuite (DDM)
DRDA (Distributed Relational Database Architecture) 7
ieșire db2drdat 139

H

hardware
performanță de rețea 126

I

IBM WebSphere
privire generală 16
informații diagnoză
privire generală 133
InfoSphere Federation Server
privire generală 6
instrucțiune COMMIT
legare statică 91
instrucțiune DESCRIBE
instrucțiuni SQL compuse 91
performanță cu instrucțiunea PREPARE 91
instrucțiune EXECUTE IMMEDIATE
proiectare aplicație 91
instrucțiune PREPARE
efect performanță 91
proiectare aplicație 91
instrucțiune ROLLBACK
legare statică 91
instrucțiune SELECT
actualizabil 91
FOR FETCH ONLY activ 91
proiectare aplicație 91
instrucțiune SET CURRENT PACKAGESET 43
instrucțiuni SQL
ajutor
afișare 161
clauza FOR FETCH ONLY din SELECT 91
COMMIT 91
DB2 Connect 5
DESCRIBE 91
EXECUTE IMMEDIATE 91
PREPARE 91
ROLLBACK 91
SELECT 91
interfață de nivel apel (CLI)
aplicații
parametru de configurare CLI/ODBC
CURRENTPACKAGESET 43
conexiuni de încredere 37
privire generală 127
istoric de evaluare politică 68
istorice
evaluare politică 68

Î

îndrumare
depanare 165
determinarea problemei 165
listă 165
Visual Explain 165

J

Java
server de aplicații
API-uri 17
DB2 Connect 17
JDBC 17
SQLJ 17
jetoane
SQLCODE-uri 57

L

legare
aplicații 47
autoritate 47
pachete
DB2 Connect 47
utilitare
DB2 Connect 47
listă de adrese puse în cache 104
listă de legătură
DB2 Connect 47

M

manager punct sync (SPM)
parametri de configurare
valori implicite 53
scenariu 52
manageri de tranzație XA
concentratori de conexiuni 97
privire generală 21
memorie
unelte pentru utilizare 85
mesaje de eroare
DB2 Connect 151
Microsoft Windows
aplicații 11
modelul X/Open DTP (distributed transaction processing)
privire generală 21
monitorizare
conexiuni 61
Windows Performance Monitor 61
monitorizarea sistemului de bază de date
clienți la distanță 61
privire generală 5
Monitorizări de procesare tranzație
OLTP 21
monitorizări procesare tranzație
actualizare multilocație 51
DB2 Connect 21
exemple 21
Tuxedo 21
monitorul de sănătate
DB2 pentru z/OS 68
mutarea datelor
DB2 Connect 55

N

noduri
directoare
actualizare 27
valori 28

noduri (*continuare*)
 nume
 foaie de lucru pentru personalizarea directorului 33
 valori bază de date sistem 27
 valori director nod 28
 NOT ATOMIC compus SQL
 proiectare aplicație 91
 NULLID 47
 nume bază de date destinație AS 29
 nume destinație simbolică
 sensibilitate la majuscule 28

O

obiect EXTNAM 140
 obiect SRVNAM 140
 obiecte de alertă
 vizualizare 74
 observații 167
 ODBC
 acces optimizat 89
 Ajustarea performanței aplicației CLI/ODBC 127
 aplicații
 parametru de configurare CLI/ODBC
 CURRENTPACKAGESET 43
 interfețe 11
 oprire
 monitorul de sănătate DB2 for z/OS 69
 opțiunea de monitor SHOW DETAIL 64

P

pachete
 servere de bază de date gazdă 47
 servere de bază de date System i 47
 parametru enableSeamlessAcr 111
 parametri BSDS (bootstrap data set)
 z/OS 28
 parametri de configurare
 agentpri 119
 dir_cache 119
 max_coordagents
 detalii 97
 privire generală 95
 MAXDARI 119
 num_initagents 95,97
 num_poolagents 95,97
 numdb 119
 rqrioblk 119
 TCP_KEEPALIVE 81
 parametrii
 directoare 33
 PRDID 140
 SYSPLEX 29
 șiruri 34
 parametru D (deconectat) 29
 parametru de configurare manager bază de date maxagents
 depreciat 119
 parametru de configurare numdb
 DB2 Connect 119
 parametru de configurare pentru suport cache de director
 Ajustarea DB2 Connect 119
 parametru de configurare rqrioblk
 ajustare 119
 parametru de configurare sistem de operare TCP_KEEPALIVE 81
 parametru dir_cache 119

parametru enableAcr 111
 parametru enableWLB 108
 parametru INTERRUPT_ENABLED (deconectat) 29
 parametru LOCALDATE 29
 parametru maxRefreshInterval 108
 parametru maxTransportIdleTime 108
 parametru maxTransports 108
 parametru maxTransportWaitTime 108
 parametru NOMAP
 dezactivare mapare SQL 57
 mapare SQL CODE 29
 parametri director DCS 57
 parametru PRDID 140
 parametru de configurare manager bază de date agentpri 119
 parametru de configurare manager bază de date max_coordagents
 detalii 97
 privire generală 95
 parametru de configurare manager bază de date num_initagents
 configurare pool de agenți inactivi 95
 privire generală 97
 parametru de configurare manager bază de date num_poolagents
 configurare pool de agenți inactivi 95
 privire generală 97
 parole
 modificare
 z/OS 43
 performanța
 acces ODBC 89
 concentrator de conexiuni 101
 DB2 Connect
 ajustare 83
 depanare 122
 privire generală 85
 rate de transfer în creștere 123
 hardware de rețea 126
 impact procesor linie de comenzi (CLP) 91
 proiectare aplicație 91
 punerea conexiunilor în pool 101
 resurse de sistem 122
 z/OS 123
 pool de transport 107
 pornire
 monitorul de sănătate DB2 for z/OS 69
 predicate
 performanță de logică 91
 proceduri memorate
 privire generală 18
 procesorul de linie de comenzi (CLP)
 instrucțiuni SQL 5
 performanța 91
 programare CGI (interfață gateway obișnuită) 15
 programarea CGI (Common Gateway Interface)
 avantaje 15
 limitări 15
 protocol de autentificare Kerberos
 DB2 Connect 41
 OS/390 43
 z/OS 43
 punerea conexiunilor în pool
 comparație concentrator conexiuni 101
 gestionarea conexiunilor 95
 privire generală 95

R

RECEIVE BUFFER 139

- referințe
 - definirea intrărilor bazei de date multiple 33
- relații de încredere
 - DB2 Connect 37
- rerutare automată client
 - configurare 111
 - detalii 79
 - pe partea client 110
 - setare 79
- rerutare client automat
 - defectare conexiune 81
- resurse de sistem
 - conflict 122
- rețele
 - ajustare 121
 - rate de transfer de date 126
 - unelte de performanță 85

S

- scalare fereastră
 - extensii RFC-1323 124
- scenariu
 - Securitate TCP/IP 44
- securitate
 - DB2 Connect 37
 - indicații 43
 - instrucțiunea GRANT 44
 - Kerberos 43
 - sugestii 43
 - suport coduri extins pentru DB2 for z/OS 43
 - TCP/IP 44
 - tipuri 33
 - valori director nod 28
- server de aplicații
 - DB2 Connect 18
 - definiție DRDA 7
- servere
 - aplicație
 - DB2 Connect 18
- servere de conectivitate
 - DB2 Connect Enterprise Edition 14
- servere web
 - DB2 Connect 18
- setare db2dsdriver AFFINITY_LIST 114
- setare db2dsdriver ALTERNATE_SERVER_LIST 114
- setare db2dsdriver CLIENT_AFFINITY 114
- sfârșit unitate de mesaje de răspuns lucru (ENDUOWRM) 140
- SOCKS
 - noduri
 - variabile de mediu obligatorii 28
- solicitanți aplicație (AR)
 - definiție DRDA 7
 - parametrii 33
- SQL
 - dinamic 91
 - static 91
- SQL dinamic
 - parametru de configurare CLI/ODBC
 - CURRENTPACKAGESET 43
 - performanța
 - tehnici 91
 - procesarea efectelor 5
- SQL static
 - performanța 91
 - procesarea efectelor 5

- SQL_ATTR_
 - TRUSTED_CONTEXT_PASSWORD
 - comutare utilizatori pe o conexiune de încredere prin CLI 39
 - TRUSTED_CONTEXT_USERID
 - comutare utilizatori pe o conexiune de încredere prin CLI 39
 - USE_TRUSTED_CONTEXT
 - creare conexiuni de încredere prin CLI 38
- SQLCA
 - buffer-e de date 139
 - câmp SQLCODE 139
- SQLCODE
 - câmp în SQLCA 139
 - fișier de mapare 57
 - Mapare 57
- SQLDA
 - dimensiune de alocare 91
- SQLSTATE
 - coduri de clasă 57
- stare sistem
 - comanda GET SNAPSHOT 62
- sumare alerte
 - vizualizare 72
- suport bidirecțional CCSID
 - parametru BIDI 29
- Sysplex
 - cerințe de configurare 105
 - informații cu prioritate 104
 - încărcare echilibrare 104
 - limitări pentru suportul pe partea client 117
 - parametru 29
 - suport client 107
 - suport DB2 Connect 103
 - System z 103
 - toleranță pagină lipsă 104
 - utilizare 104

Ș

- șiruri de parametri
 - virgulă dublă 29
 - virgule 29

T

- TCP/IP
 - comanda ACCSEC 140
 - comanda SECCHK 140
 - configurare
 - conexiuni gazdă 12
 - DOMAIN 28
 - extensii RFC-1323 124
 - nume de gazdă de la distanță 28, 33
 - nume gazdă 33
 - nume servicii 28
 - numere port 33
 - port resynch 28
 - RESPORT 28
 - scenarii de autentificare 44
 - securitate 43
 - TCPPOPT 28
- termeni și condiții
 - publicații 166
- testarea
 - actualizare multilocație 52
- timpi de răspuns
 - DB2 Connect 85

- tip autentificare PROGRAM 44
- tip autentificare SAME 44
- tip de autentificare CLIENT
 - DB2 Connect 41
- tip de autentificare DATA_ENCRYPT 41
- tip de autentificare SERVER
 - DB2 Connect 41
- tip de autentificare SERVER_ENCRYPT
 - DB2 Connect 41
- tip de date CHAR
 - detalii 125
- tip de date INTEGER
 - conversie de date gazdă 125
- tip de date pachet zecimal 125
- tip de date VARCHAR
 - privire generală 125
- tip de date zecimal zonat 125
- tipul de autentificare SERVER_ENCRYPT_AES 41
- tipuri de autentificare NONE 44
- tipuri de date
 - caracter 125
 - CHAR 125
 - conversie
 - efect performanță 125
 - INTEGER
 - conversie de date gazdă 125
 - pachet zecimal 125
 - VARCHAR
 - privire generală 125
 - virgulă flotantă
 - conversie de date gazdă 125
 - zecimal zonat 125
- tipuri de date caracter 125
- tipuri de date virgulă flotantă
 - conversie 125
- transporturi 107
- tranzacții
 - actualizare multilocație 7, 51
 - aplicații distribuite XA 54
 - comitere în două faze 7
 - cuplat slab
 - DB2 Connect 54
 - DB2 Connect Enterprise Edition 21
 - debit
 - DB2 Connect 85
 - distribuită 51
 - unitate de lucru (UOW) 7
- Tranzacții
 - Monitorizări de procesare tranzacție 21
- Tuxedo
 - DB2 Connect Enterprise Edition 21

U

- unelte
 - performanța 85
 - utilizare CPU 85
 - utilizare memorie 85
- unitate de lucru (UOW)
 - distribuită 51
 - la distanță 8
 - privire generală 7
- unități de lucru distribuite
 - actualizare multilocație 51
 - comitere în două faze 51
 - privire generală 7
 - servere suportate 51

- unități de lucru la distanță
 - caracteristici 8
 - exemplu 8
 - privire generală 8
- urmăriri
 - date între DB Connect și server 139
 - DB2 135, 136
 - DB2 Connect 135
 - DRDA
 - exemple 142
 - informații buffer 146
 - interpretare 139
 - exemple de fișiere de ieșire 142
 - Fișier de ieșire 139, 140
- utilitar ddcstrc 140
- utilitar de export
 - transfer de date între gazde și stații de lucru 55
- utilitar de import
 - transferarea datelor între gazdă și stația de lucru 55
- utilitar de stare de proces
 - comandă 133, 140
- utilitar de urmărire (db2drdat) 139
- utilitare
 - administrare DB2 Connect 5
 - db2drdat 139
 - ddcspkgn 47
 - legare 47
 - monitorizarea sistemului de bază de date 5
 - ps (process status) 133, 140
 - urmărire 139

V

- vrajitor de configurare a actualizării multilocației 52
- vrajitori
 - actualizare multilocație 52

W

- WebSphere
 - privire generală 16
- WebSphere MQ
 - DB2 Connect 101
- Windows
 - Performance Monitor
 - monitorizare aplicații DB2 61

X

- XA
 - conexiuni de încredere 37
 - exemple de concentrator 97
 - manageri de resurse 21



Tipărit în S.U.A.

SA22-1403-01



Spine information:

IBM DB2 Connect 9.7 **Versiune 9 Ediție 7**

Ghidul utilizatorului DB2 Connect

