

Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à l'Annexe B, «Remarques», à la page 191.

Remarque

Certaines illustrations de ce manuel ne sont pas disponibles en français à la date d'édition.

Première édition - juillet 2009

Réf. US : SC27-2434-00

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.can.ibm.com> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France
Direction Qualité
Tour Descartes
92066 Paris-La Défense Cedex 50*

© Copyright IBM France 2009. Tous droits réservés.

© **Copyright International Business Machines Corporation 1993, 2009.**

Table des matières

Avis aux lecteurs canadiens vii

A propos de ce manuel ix

Partie 1. Concepts de DB2 Connect 1

Chapitre 1. DB2 Connect 3

Présentation de l'offre produit DB2 Connect 3

Fonctions fournies avec DB2 Connect version 8. 3

Bases de données hôte 4

DB2 Connect et instructions SQL 5

Utilitaires d'administration DB2 Connect 6

InfoSphere Federation Server et DB2 Connect 7

Chapitre 2. Architecture de base de données relationnelle répartie (DRDA). 9

DRDA et accès aux données 9

DB2 Connect et DRDA 10

Unité d'oeuvre éloignée 11

Requêtes réparties 13

Chapitre 3. Scénarios DB2 Connect 15

Accès direct aux bases de données hôte 15

Accès à l'hôte System z ou aux données IBM i DB2

à l'aide de DB2 Connect Personal Edition 17

Produits serveur DB2 Connect en tant que serveurs

de connectivité 19

DB2 Connect et applications Web 20

DB2 Connect et IBM WebSphere 21

DB2 Connect en tant que serveur d'applications

Java 22

DB2 Connect sur le serveur Web 23

DB2 Connect et serveurs d'applications 23

DB2 Connect et moniteurs de traitement de

transactions 27

Partie 2. Référence pour DB2 Connect. 31

Chapitre 4. Mise à jour des répertoires de bases de données 33

Valeurs du répertoire système des bases de données 33

Valeurs du répertoire des noeuds 34

Valeurs du répertoire DCS 35

Feuille de travail de personnalisation du répertoire

Définition d'entrées multiples pour la même base de

données 40

Traitement des données bidirectionnelles (BiDi) 41

Chapitre 5. Sécurité de DB2 Connect 45

Connexions sécurisées via DB2 Connect 45

Création et arrêt d'une connexion sécurisée à l'aide de CLI 46

Changement d'utilisateurs sur une connexions sécurisée via CLI 48

Remarques sur l'authentification DB2 Connect. 50

Support Kerberos 51

Conseils et astuces relatifs à la sécurité z/OS 52

Types d'authentification pris en charge avec DB2

Connect 53

Chapitre 6. Liaison d'applications et d'utilitaires (DB2 Connect) 55

Chapitre 7. Mises à jour multisite 59

Activation des mises à jour multisite à l'aide du Centre de contrôle 60

Test de mise à jour multisite à l'aide du Centre de contrôle 60

Mise à jour multisite et gestionnaire de points de synchronisation 61

Configuration de DB2 Connect avec un gestionnaire de transactions compatible XA 62

Prise en charge par DB2 Connect des transactions à couplage lâche 62

Chapitre 8. Déplacement de données avec DB2 Connect. 63

Chapitre 9. Mappage SQLCODE 67

Désactivation du mappage SQLCODE 67

Personnalisation du mappage SQLCODE 67

Chapitre 10. Surveillance du système de base de données et DB2 Connect. 73

Contrôle des connexions des clients éloignés 73

Contrôle des performances à l'aide du moniteur de

performances de Windows 73

Utilisation des commandes GET SNAPSHOT 74

Etat de l'application DCS. 76

Moniteur d'état de santé et alertes. 81

Moniteur de santé DB2 for z/OS - Généralités. 81

Démarrage, arrêt et régénération du moniteur de

santé DB2 for z/OS. 82

Affichage, soumission et sauvegarde des actions

recommandées 83

Affichage des récapitulatifs des alertes de santé 86

Affichage des objets alerte de santé 87

Partie 3. Haute disponibilité et DB2 Connect. 89

Chapitre 11. Haute disponibilité et équilibrage de la charge de travail pour la connectivité de la base de données hôte	91
--	-----------

Chapitre 12. Configuration et description de la redirection client automatique (DB2 Connect).	93
--	-----------

Chapitre 13. Configuration de la redirection automatique du client pour la technologie de distributeur de connexion client	95
---	-----------

Partie 4. Réglage et DB2 Connect	97
---	-----------

Chapitre 14. DB2 Connect remarques sur les performances	99
--	-----------

Chapitre 15. Optimisation de l'accès ODBC.	103
---	------------

Chapitre 16. Conception d'application	105
--	------------

Chapitre 17. Gestion des connexions	109
Regroupement de connexions	109
Concentrateur de connexion	111
Regroupement et concentrateur de connexions	116
Un concentrateur de connexion est requis avec WebSphere MQ Transaction Manager et DB2 for z/OS	117

Chapitre 18. Prise en charge de Sysplex par le serveur DB2 Connect	119
Considérations concernant l'exploitation de SYSPLEX sur System z	119
Exploitation de Sysplex avec DB2	120
Configuration requise pour Sysplex	121

Chapitre 19. Support d'un Sysplex client	123
Équilibrage de charge au niveau des transactions (côté client)	123
Configuration de l'équilibrage de charge au niveau des transactions (côté client)	125
Redirection automatique du client (côté client)	127
Configuration de la redirection automatique du client (côté client)	128
Support XA (côté client)	131
Activation du support XA (côté client)	132
Configuration des affinités client	133
Limitations pour l'utilisation du support du Sysplex client	138

Chapitre 20. Optimisation de DB2 Connect.	139
--	------------

Optimisation de la base de données hôte	141
Considérations d'optimisation réseau	141
Conflit de ressources système	143
Résolution des incidents de performances de DB2 Connect	143
Optimisation de DB2 for z/OS	144
Augmentation des débits de transfert des données de DB2 Connect	144
Bloc de requête supplémentaire	144
Mise à l'échelle des fenêtres RFC-1323	146
Conversion de données sur l'hôte	146
Types de données pour les données de type caractères.	147
Matériel réseau.	147

Chapitre 21. Optimisation des performances d'applications CLI/ODBC	149
---	------------

Partie 5. Identification des incidents.	151
--	------------

Chapitre 22. Identification et résolution des incidents DB2 Connect	153
Collecte d'informations pertinentes	153
Connexion initiale non aboutie	153
Incidents rencontrés après une connexion initiale	154
Outils de diagnostic	156

Chapitre 23. Traces DB2 dans DB2 Connect.	157
Obtention d'une trace DB2 avec db2trc	157
Vidage d'un fichier de trace DB2	158
Formatage d'un fichier de trace DB2.	159

Chapitre 24. Fichiers de trace DRDA	161
Utilitaire de trace	161
Sortie de trace	162
Analyse du fichier de sortie de trace.	163
Modèles de fichiers de sortie de trace	164
Informations de mémoire tampon postérieures pour les traces DRDA	169

Partie 6. Messages	171
-------------------------------------	------------

Chapitre 25. Incidents DB2 Connect courants	173
--	------------

Partie 7. Annexes.	177
-----------------------------------	------------

Annexe A. Présentation des informations techniques DB2	179
Bibliothèque technique DB2 au format PDF ou en version papier	180
Commande de manuels imprimés DB2.	182

Affichage de l'aide sur les codes d'état SQL à partir de l'interpréteur de commandes	183
Accès aux différentes versions du centre de documentation DB2	184
Affichage des rubriques dans votre langue préférée dans le centre de documentation DB2	184
Mise à jour du centre de documentation DB2 installé sur votre ordinateur ou sur votre serveur intranet	185
Mise à jour manuelle du centre de documentation DB2 installé sur votre ordinateur ou sur votre serveur intranet	186

Tutoriels DB2	188
Informations relatives à la résolution d'incidents sur DB2	188
Dispositions	189

Annexe B. Remarques 191

Index 195

Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.








OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
 (Pos1)		Home
Fin	Fin	End
 (PgAr)		PgUp
 (PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
 (Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

A propos de ce manuel

Le manuel *DB2 Connect - Guide d'utilisation* contient toutes les informations dont vous avez besoin pour comprendre et utiliser le produit DB2 Connect. Il présente les concepts relatifs à DB2 Connect avec des scénarios standard indiquant les relations entre DB2 Connect et les autres parties de l'environnement réseau. Il traite des répertoires de bases de données, de la sécurité entre les systèmes, des mises à jour multi-sites et de la surveillance de DB2 Connect. Il explique comment DB2 Connect gère la haute disponibilité dans votre environnement réseau. Il indique comment préserver un bon niveau de performances de DB2 Connect et dans tout le réseau. Certaines rubriques traitent de l'identification et de la résolution des incidents.

A qui s'adresse ce manuel ?

Aux administrateurs système, administrateurs de base de données, spécialistes des communications et installateurs de logiciels.

Partie 1. Concepts de DB2 Connect

Chapitre 1. DB2 Connect

DB2 Connect permet une connectivité rapide et fiable vers les bases de données grand système IBM® dans les domaines de l'e-business et d'autres applications s'exécutant sur les systèmes d'exploitation Linux®, UNIX® et Windows®.

DB2 Connect Personal Edition fournit une connectivité directe avec les serveurs System z et IBM Power Systems, tandis que les produits serveur DB2 Connect offrent une connectivité indirecte permettant aux clients d'accéder aux serveurs System z et IBM Power Systems via la passerelle DB2 Connect. Divers serveurs DB2 Connect offrent des solutions de conditionnement et de licence permettant la sélection d'un produit adapté à votre environnement.

Présentation de l'offre produit DB2 Connect

DB2 Connect dispose de plusieurs solutions de connexion (notamment DB2 Connect Personal Edition) et de différents produits serveur 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

Pour plus d'informations sur l'offre produit DB2 Connect, consultez www.ibm.com/software/data/db2/db2connect/

Fonctions fournies avec DB2 Connect version 8

Cette section présente un récapitulatif des améliorations apportées dans DB2 Connect version 8. Pour consulter la liste des modifications intégrées dans DB2 version 9 et affectant le fonctionnalité DB2 Connect, reportez-vous aux rubriques suivantes :

- Récapitulatif du groupe de correctifs DB2 Connect version 9.5
- Récapitulatif du groupe de correctifs DB2 Connect version 9.1

Fonctions fournies avec DB2 Connect Version 8 Release 2

DB2 Connect version 8.2 inclut les améliorations suivantes :

- Redirection automatique des clients

Si une connexion TCP/IP établie avec un serveur DB2 Connect est perdue, le client tente automatiquement d'établir à nouveau la connexion si un serveur de remplacement existe. Le serveur de remplacement est spécifié sur l'instance de serveur et son emplacement est envoyé au client lors de la connexion.

- Chiffrement des données

La communication client/serveur fournit désormais le chiffrement des données utilisateur car elle sillonne le réseau.

Fonctions fournies avec DB2 Connect Version 8 Release 1 (notamment tous les groupes de correctifs et les niveaux de modification)

DB2 Connect version 8.1 comporte les améliorations suivantes :

- Support d'instructions SQL plus longues (jusqu'à 2 Mo)
Les instructions SQL jusqu'à 2 Mo peuvent parcourir les applications CLI et JDBC. Cependant, l'interface incorporée reste limitée à 64 K.
- Les données de diagnostic qui identifient l'origine d'une instruction SQL
Permettent de déterminer quel programme d'application a émis une instruction spécifique présente dans le cache d'instructions SQL dynamiques DB2 for z/OS.
- Tableau d'entrée au niveau des colonnes
Permet aux applications de fournir divers ensembles de paramètres pour une même instruction SQL.
- Contrôle du temps réseau
De nouveaux éléments de contrôle sont utilisés afin d'avoir une meilleure idée de l'activité de la base de données et du trafic réseau au niveau de l'application ou de la base de données.
- Support de curseur flottant dynamique DB2 CLI
Les curseurs flottants dynamiques sont désormais pris en charge dans DB2 CLI serveurs DB2 Universal Database (UDB) pour z/OS version 8.1 ou version ultérieure.
- Support eWLM
Offre la fonction de contrôle des unités d'oeuvre de bout en bout via des groupes de logiciels intermédiaires afin de déterminer les goulots d'étranglement.
- Améliorations apportées à la commande DB2 ping
La commande DB2 ping prend désormais en charge la spécification d'une taille de paquet de demandes et réponses.

Remarque : DB2 Connect ne prend pas en charge la commande PING lorsqu'elle est exécutée d'un client version 7 vers l'hôte par l'intermédiaire d'une passerelle version 9.

Bases de données hôte

Le terme *base de données* est utilisé tout au long du document pour décrire un système de gestion de base de données relationnelle (RDBMS). D'autres systèmes avec lesquels DB2 Connect communique peuvent utiliser le terme "base de données" pour décrire un concept quelque peu différent. Le terme DB2 Connect "base de données" peut également désigner :

System z

DB2 for z/OS. Un sous-système DB2 for z/OS identifié par sa propriété LOCATION NAME. Le LOCATION NAME peut être déterminé lorsque vous vous connectez au TSO et que vous exécutez la requête SQL à l'aide de l'un des outils de requête disponibles :

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

Le LOCATION NAME est également défini dans l'ensemble de données d'amorce (BSDS) ainsi que dans le message DSNL004I (LOCATION=location), qui est écrit lorsque l'utilitaire DDF (Distributed Data Facility) démarre. Le LOCATION NAME prend en charge jusqu'à

8 alias de noms d'emplacement, ce qui permet aux applications d'utiliser des noms dbalias différents pour accéder à un serveur z/OS version 8. Utilisez la commande z/OS -display ddf pour obtenir le nom de l'emplacement du serveur DB2, le nom de domaine, l'adresse IP et le port.

- VSE** DB2 for VSE fonctionnant sur une partition de base de données identifiée par son DBNAME
- VM** DB2 for VM fonctionnant sur une machine virtuelle CMS identifiée par son DBNAME

Serveurs IBM Power Systems

DB2 for IBM i, qui est partie intégrante du système d'exploitation IBM i. Une seule base de données peut exister sur un système IBM Power Systems à moins que le système ne soit configuré pour utiliser des pools de stockage auxiliaire indépendants.

DB2 Connect et instructions SQL

DB2 Connect fait suivre les instructions SQL soumises par des programmes d'application aux serveurs de base de données grand système IBM.

DB2 Connect peut transférer presque toutes les instructions SQL valides ainsi que les interfaces de programmation DB2 prises en charge :

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

Prise en charge du SQL imbriqué

Il existe deux types de traitement SQL imbriqué : le SQL statique et le SQL dynamique. Le SQL statique réduit le temps nécessaire à l'exécution d'une instruction SQL en la traitant à l'avance. Le SQL dynamique est traité lorsque l'instruction SQL est soumise au serveur de base de données grand système IBM. Le SQL dynamique est plus flexible mais potentiellement plus lent. La décision d'utiliser le SQL statique ou dynamique revient au programmeur d'application. Deux types sont pris en charge par DB2 Connect.

L'implémentation de SQL n'est pas la même selon les différents serveurs de base de données grand système IBM. DB2 Connect prend entièrement en charge les implémentations courantes d'IBM SQL, ainsi que les implémentations SQL pour DB2 for z/OS, DB2 Server for VM and VSE (anciennement SQL/DS) et DB2 for IBM i. IBM SQL est fortement recommandé pour la gestion de l'indépendance des bases de données.

Utilitaires d'administration DB2 Connect

Important : Le Centre de contrôle et les composants associés sont devenus obsolètes dans la version 9.7 et seront supprimés dans une version ultérieure. Pour plus d'informations, voir la rubrique «Outils du Centre de contrôle et Serveur d'administration DB2 (DAS) devenus obsolètes» du manuel *Nouveautés de la version 9.7*.

Les utilitaires suivants sont disponibles pour aider les administrateurs DB2 Connect :

- L'Interpréteur de commandes (CLP) vous permet d'émettre des instructions SQL sur une base de données de serveur grand système IBM. Il suit les instructions SQL vers la base de données que vous avez spécifiée.
- Le centre de commande DB2 fournit une interface graphique à l'Interpréteur de commandes (CLP).
- Les utilitaires d'importation et d'exportation vous permettent de charger, d'importer et d'exporter des données depuis et vers un fichier situé sur un poste de travail et une base de données de serveur grand système IBM. Ces fichiers peuvent être utilisés pour importer des données dans des bases de données, des feuilles de calcul et d'autres applications fonctionnant sur un poste de travail.
- Si vous utilisez un serveur DB2 Connect, vous pouvez utiliser l'observateur d'événements ou le moniteur de performances. Grâce à l'observateur d'événements, vous pouvez visualiser les événements exceptionnels consignés par DB2 Connect. Grâce au moniteur de performances, vous pouvez contrôler et gérer les performances des serveurs DB2 Connect, localement ou à distance.
- Le centre de contrôle DB2 vous permet d'administrer ou de contrôler tous les aspects des serveurs DB2 Connect. Il permet également aux administrateurs de travailler avec des objets de base de données DB2 for z/OS, tels que des tables, des vues, des pools de mémoire tampon et des unités d'exécution.
- L'utilitaire moniteur du gestionnaire de bases de données permet à l'administrateur du système de contrôler les connexions système. Cette fonction est uniquement disponible lorsque DB2 Connect agit en tant que serveur. Cet utilitaire aide également l'administrateur système à déterminer l'origine d'une erreur. L'administrateur système peut mettre en corrélation des applications client avec les travaux correspondants s'exécutant sur le serveur de base de données grand système IBM.

Remarque : Dans les versions précédentes, les outils d'administration graphiques DB2, tels que le centre de contrôle, étaient pris en charge sur toutes les plateformes. A partir de la version 9, les outils d'administration graphiques DB2 sont uniquement pris en charge sous Windows x86, Windows x64 (AMD64/EM64T), Linux sur x86, et Linux sur AMD64/EM64T. Pour toutes les plateformes, vous pouvez utiliser l'interpréteur de commandes DB2 (CLP) à des fins d'administration.

InfoSphere Federation Server et DB2 Connect

InfoSphere Federation Server est une offre de produit distincte qui fournit l'accès à des données spécifiques et permet de les intégrer via diverses sources de données multiconstructeur alors que DB2 Connect rend possible l'optimisation de grands volumes de données situés sur des hôtes et des serveurs de milieu de gamme existants.

InfoSphere Federation Server facilite l'intégration des informations en autorisant l'affichage et la manipulation d'un ensemble de sources de données comme s'il s'agissait d'une même source. L'accès aux sources de données est ainsi totalement transparent pour l'application appelante. InfoSphere Federation Server, qui fonctionne conjointement aux produits serveur DB2 Connect, InfoSphere Federation Server permet un accès natif en lecture et en écriture à la famille de produits DB2, ainsi qu'aux bases de données Informix, Oracle, Sybase, Teradata et Microsoft® SQL Server. InfoSphere Federation Server permet également un accès en lecture aux sources de données non relationnelles et de sciences biologiques, telles que Documentum, IBM Lotus Extended Search, aux fichiers structurés en tableaux et XML. Vous pouvez l'utiliser pour formuler des requêtes concernant les données d'un système fédéré.

Chapitre 2. Architecture de base de données relationnelle répartie (DRDA)

L'architecture de base de données relationnelle répartie (DRDA) est un ensemble de protocoles qui permet à plusieurs systèmes de base de données, IBM et non IBM, ainsi qu'à des programmes d'application de fonctionner ensemble. Toute combinaison de produits de gestion de base de données relationnelle qui utilise DRDA peut être connectée pour former un système de gestion de base de données relationnelle répartie. DRDA coordonne la communication entre des systèmes en définissant les éléments qui peuvent ou non être échangés.

Unité d'oeuvre

Une *unité d'oeuvre (UOW)* est une seule transaction logique. Elle consiste en une séquence d'instructions SQL dans laquelle toutes les opérations sont effectuées avec succès ou dans laquelle la séquence est considérée comme un échec dans son ensemble.

Unité d'oeuvre répartie

Une *unité d'oeuvre répartie (DUOW)*, également connue sous le nom de mise à jour multisite, implique plus d'un serveur de base de données au sein d'une unité d'oeuvre. Une DUOW possède les caractéristiques suivantes :

- Plusieurs serveurs de gestion de base de données sont mis à jour par unité d'oeuvre.
- L'application dirige la répartition du travail et initie les validations.
- Plusieurs requêtes peuvent se trouver dans une unité d'oeuvre.
- Un serveur de gestion de base de données est dédié à chaque requête.
- La validation est coordonnée à travers plusieurs serveurs de base de données.

DRDA et accès aux données

Bien que DRDA définit les protocoles de communication de base de données, il ne définit pas les interfaces de programmation (ou API) que les programmeurs doivent utiliser. En règle générale, DRDA peut être utilisé par un programme d'application pour transmettre toute requête pouvant être exécutée par un serveur DRDA cible. Tous les serveurs DRDA disponibles à l'heure actuelle, peuvent exécuter les requêtes SQL transférées par un programme d'application via DB2 Connect.

IBM fournit aux programmeurs d'application des outils de génération de requêtes SQL pour les systèmes d'exploitation Windows, UNIX et Linux. Ces outils sont des composants du client DB2. Le gestionnaire de base de données DB2 prend en charge plusieurs interfaces de programmation : ADO.NET, JDBC, SQLJ, PHP, Perl DBI, SQL imbriqué, DB2 Call Level Interface (DB2 Call Level Interface) et OLE DB. Ces API peuvent être utilisées par les programmeurs afin de générer des applications en plusieurs langues.

DB2 Connect et DRDA

DB2 Connect implémente l'architecture DRDA afin de diminuer les coûts et la complexité de l'accès aux données stockées sur DB2 for IBM i, DB2 for IBM Power Systems, DB2 for z/OS, DB2 Server for VM and VSE, et d'autres serveurs de base de données compatibles DRDA. Grâce à l'exploitation intégrale de l'architecture DRDA, DB2 Connect offre une solution performante et économique possédant les caractéristiques système que les clients attendent.

Dans la terminologie DRDA, un *demandeur d'application (AR)* est le code qui gère la fin de l'application d'une connexion répartie. L'AR est l'application qui demande les données. DB2 Connect agit en tant que demandeur d'application à la demande des programmes d'application qui peuvent être des programmes locaux situés sur le poste de travail DB2 Connect ou un client distinct situé à distance de DB2 Connect.

Un *serveur d'applications (AS)* est le code qui gère la fin de la base de données d'une connexion.

DRDA prend également en charge les connexions multiniveau entre un demandeur d'application et un serveur. Dans cette topologie, le serveur auquel un demandeur d'application se connecte est un serveur d'applications, et tout autre serveur situé plus en aval est appelé serveur de base de données (DS) car il n'interagit pas avec le demandeur d'application. En outre, afin de mettre en exergue son rôle de n'être ni le système à l'origine d'une requête de base de données, ni le système qui effectue la fonction de base de données pour la requête, tout serveur d'applications ou de base de données situé entre le demandeur d'application et le serveur de base de données final est également appelé "serveur intermédiaire". L'utilisation des serveurs de base de données et des serveurs intermédiaires est prise en charge par DB2 Connect.

La figure 1 illustre le flot de données entre le poste de travail DB2 Connect et le serveur grand système IBM au cas où sont présents uniquement des clients locaux.

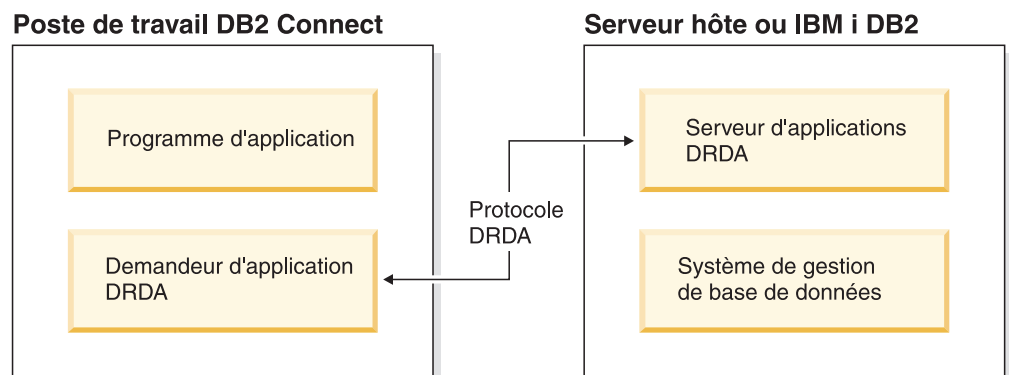


Figure 1. Flot de données entre un serveur DB2 Connect et un serveur grand système IBM

Afin de mettre en oeuvre les connexions entre les systèmes de gestion de base de données serveur DRDA et les clients IBM Data Server, DRDA utilise les architectures suivantes :

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

Ces architectures sont utilisées comme éléments d'assemblage. Les flots de données qui parcourent le réseau sont spécifiés par l'architecture DRDA qui documente un protocole de flot de données prenant en charge les accès aux bases de données relationnelles.

Une requête est dirigée vers le bon emplacement au moyen de répertoires contenant divers types d'informations de communication et le nom de la base de données du serveur DRDA à laquelle vous accédez.

Unité d'oeuvre éloignée

Une *unité d'oeuvre éloignée* permet à un utilisateur ou un à programme d'application de lire ou de mettre à jour les données d'un seul emplacement. Elle prend en charge l'accès au sein d'une même unité d'oeuvre éloignée. Un programme d'application peut mettre à jour plusieurs bases de données éloignées, mais il ne peut accéder qu'à une seule base de données au sein d'une unité d'oeuvre.

L'unité d'oeuvre éloignée possède les caractéristiques suivantes :

- Plusieurs requêtes (instructions SQL) sont prises en charge par unité d'oeuvre éloignée.
- Plusieurs curseurs sont pris en charge par unité d'oeuvre éloignée.
- Chaque unité d'oeuvre éloignée peut uniquement mettre à jour une base de données.
- Le programme d'application valide ou annule l'unité d'oeuvre éloignée. Lorsqu'une erreur se produit, le serveur de base de données ou DB2 Connect peut annuler l'unité d'oeuvre éloignée.

Par exemple, la figure 2 illustre un client de base de données exécutant une application de transfert de fonds qui accède à une base de données contenant des tables de comptes de chèque et d'épargne ainsi qu'une grille tarifaire des transactions. L'application doit :

- Accepter la somme à transférer à partir de l'interface utilisateur.
- Soustraire la somme du compte d'épargne et déterminer le nouveau solde de compte.
- Lire la grille tarifaire afin de déterminer les frais de transaction d'un compte d'épargne avec le solde de compte donné.
- Soustraire les frais de transaction du compte d'épargne.
- Ajouter la somme du transfert au compte courant.
- Valider la transaction (unité d'oeuvre).

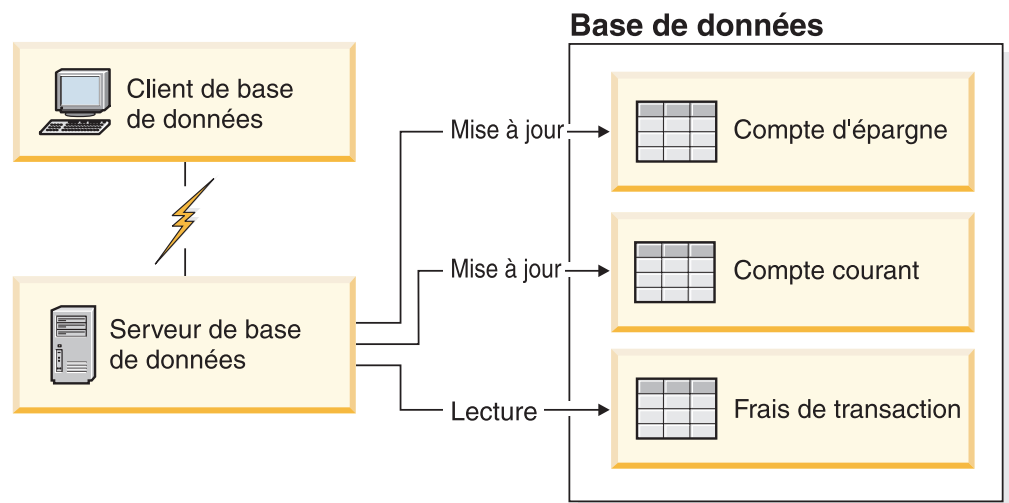


Figure 2. Utilisation d'une seule base de données au cours d'une transaction

Pour définir une telle application, vous devez :

1. Créer des tables pour le compte d'épargne, le compte courant et la grille tarifaire des transactions dans la même base de données.
2. S'il est physiquement éloigné, définir le serveur de base de données de sorte qu'il utilise le protocole de communication approprié.
3. S'ils sont physiquement éloignés, cataloguer le noeud et la base de données afin d'identifier la base de données sur le serveur de base de données.
4. Précompiler votre programme d'application pour spécifier une connexion de type 1, c'est-à-dire, spécifier CONNECT(1) dans la commande PREP.

Requêtes réparties

Une *requête répartie* est une fonction de base de données répartie qui permet aux applications et aux utilisateurs de soumettre des instructions SQL référençant deux ou plusieurs SGDB ou bases de données dans une même instruction. Par exemple, une jointure entre tables de deux sous-systèmes DB2 for z/OS différents.

DB2 Connect prend en charge les requêtes réparties dans les bases de données et les SGDB. Par exemple, vous pouvez effectuer une opération UNION entre une table DB2 et une vue Oracle. Les SGDB pris en charge incluent des membres de la famille DB2 (tels que DB2 Database for Linux, UNIX, and Windows, DB2 for z/OS, et DB2 for i) et Oracle. La prise en charge multivendeur est disponible lors de l'utilisation de DB2 Connect avec InfoSphere Federation Server.

La requête répartie offre une *transparence d'emplacement* pour les objets de base de données. Si des informations (dans des tables et des vues) sont déplacées, des références vers ces informations (appelées *pseudonymes*) peuvent être mises à jour sans que les applications requérant ces informations ne soient modifiées. La requête répartie offre également une *compensation* aux SGDB qui ne prennent pas en charge tous les dialectes SQL DB2 ou certaines fonctions d'optimisation. Les opérations qui ne peuvent être effectuées avec un SGDB (tel que le SQL récursif) sont exécutées avec DB2 Connect.

La requête répartie fonctionne de manière *semi-autonome*. Par exemple, les requêtes DB2 contenant des références à des objets Oracle peuvent être soumises alors que les applications Oracle accèdent au même serveur. La requête répartie ne monopolise pas ou ne restreint pas l'accès (du point de vue de l'intégrité ou des contraintes de verrouillage aux objets Oracle ou autres objets de SGDB).

La mise en oeuvre de la fonction de requête répartie consiste en une instance DB2 Connect, en une base de données qui agira en tant que base de données fédérée et une ou plusieurs sources de données distantes. La *base de données fédérée* contient des entrées de catalogue identifiant les sources de données et leurs caractéristiques. Une *source de données* se compose d'un SGDB et de données. Les applications se connectent à la base de données fédérée en suivant le même procédé que pour n'importe quelle base de données DB2. La base de données fédérée DB2 Connect n'est pas sous licence pour la gestion des données utilisateur. Son seul objet est de contenir des informations sur les sources de données.

Lorsqu'un système fédéré est configuré, vous pouvez accéder aux informations relatives aux sources de données comme si elles se trouvaient dans une même grande base de données. Les utilisateurs et les applications envoient des requêtes à une base de données fédérée qui extrait ensuite les données des systèmes de la famille DB2 et Oracle, en cas de besoin. Les utilisateurs et les applications spécifient des pseudonymes dans les requêtes qui fournissent des références vers les tables et les vues situées dans des sources de données. Du point de vue de l'utilisateur final, les pseudonymes sont identiques aux alias.

Plusieurs facteurs peuvent affecter les performances des requêtes réparties. Le facteur le plus important consiste à garantir que des informations récentes sur les sources de données et leurs objets sont conservées dans le catalogue global des bases de données fédérées. Ces informations sont utilisées par l'optimiseur DB2 et peuvent affecter les décisions, entraînant le transfert des opérations en vue de leur évaluation dans les sources de données.

Chapitre 3. Scénarios DB2 Connect

DB2 Connect peut fournir diverses solutions pour répondre aux besoins d'accès à votre base de données grand système IBM. Cette rubrique élabore divers scénarios pouvant s'appliquer aux besoins spécifiques de votre environnement.

Accès direct aux bases de données hôte

La fonction de base de DB2 Connect est d'offrir une connexion directe à une base de données hôte depuis des applications bureautiques s'exécutant sur vos postes de travail. IBM Data Server Driver Package avec licence DB2 Connect constitue l'approche la plus simple pour fournir cette solution.

Chaque poste de travail sur lequel DB2 Connect Personal Edition est installé peut établir une connexion TCP/IP directe avec les serveurs DB2 for z/OS, DB2 for IBM i, et DB2 Database for Linux, UNIX, and Windows. En outre, les applications peuvent se connecter à plusieurs bases de données de la famille DB2 et les mettre à jour au cours de la même transaction en bénéficiant de l'intégrité complète des données offerte par le protocole de validation à deux phases.

La figure 3 illustre une connexion directe vers un serveur IBM grand système depuis un poste de travail sur lequel est installé DB2 Connect Personal Edition.

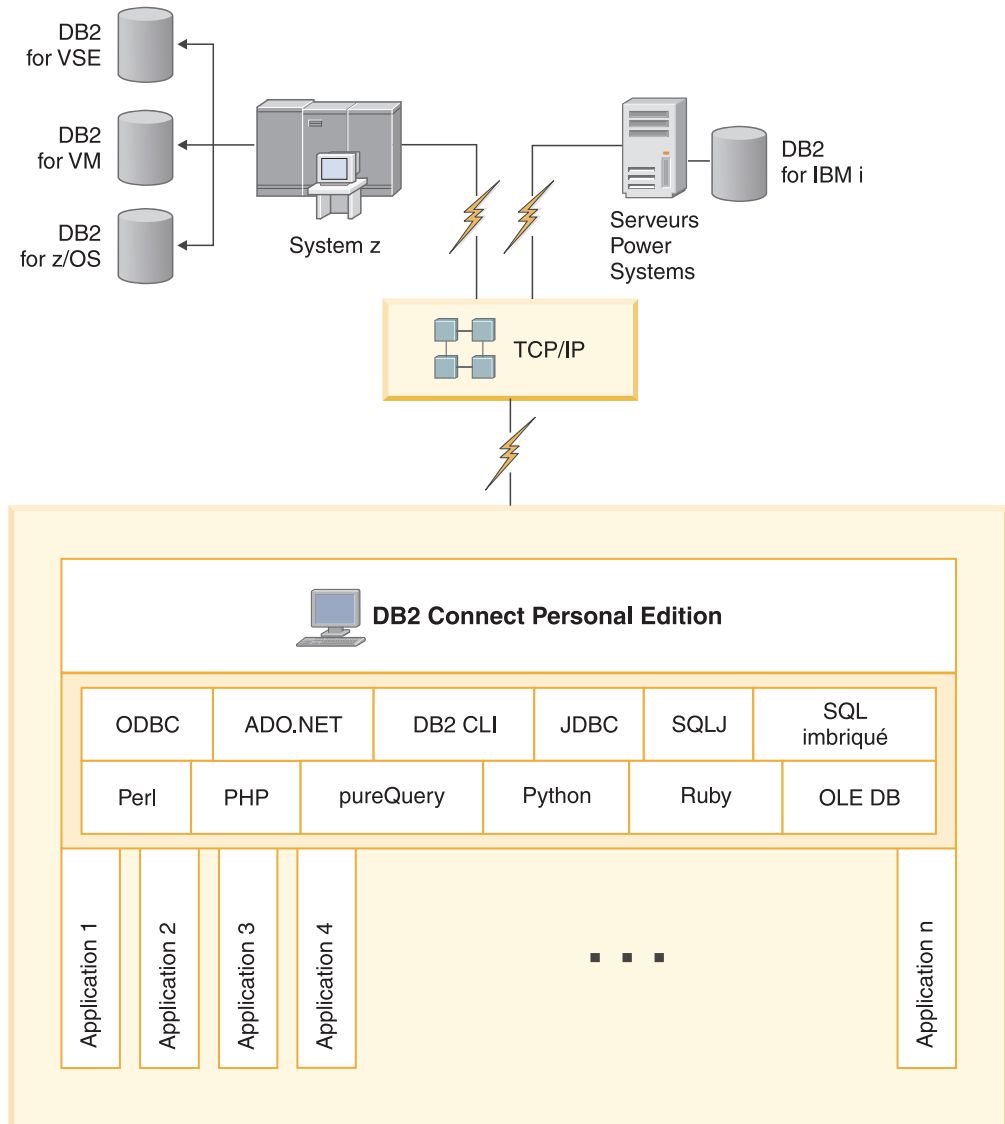


Figure 3. Connexion directe entre DB2 Connect et un serveur de base de données grand système IBM

Remarque :

1. DB2 ne doit pas être nécessairement installé sur le poste de travail DB2 Connect Personal Edition. Si vous souhaitez un système de gestion de base de données relationnelle complet sur le poste de travail DB2 Connect Personal Edition, commandez DB2.
2. Toutes les fonctionnalités IBM Data Server Client sont disponibles avec DB2 Connect Personal Edition.
3. Si la connexion est perdue avec un serveur de base de données DB2 for z/OS pour laquelle l'exploitation Sysplex est activée, le client tente automatiquement de rétablir la connexion.

Accès à l'hôte System z ou aux données IBM i DB2 à l'aide de DB2 Connect Personal Edition

Une connexion directe sans serveurs intermédiaires est une configuration très pratique qui présente de nombreux avantages, Ceci est particulièrement le cas dans les situations où le serveur de base de données grand système IBM gère la connectivité TCP/IP. Dans ces situations, chaque poste de travail DB2 Connect établit une connexion directe avec le serveur de base de données grand système IBM.

La connectivité TCP/IP nécessite que la base de données grand système IBM prenne en charge le protocole TCP/IP. Les versions suivantes prennent en charge les connexions TCP/IP natives :

- DB2 for z/OS version 7.1, ou ultérieure
- DB2 for IBM i version 5 édition 1, ou ultérieure, et
- DB2 Server for VM and VSE version 7 ou ultérieures

Pour pouvoir vous connecter à un serveur de base de données grand système IBM, vous avez besoin d'une licence DB2 Connect, laquelle peut être ajoutée à un client IBM Data Server.

La figure 4 présente un poste de travail, sur lequel DB2 Connect Personal Edition est installé, connecté directement à un serveur de base de données grand système IBM.

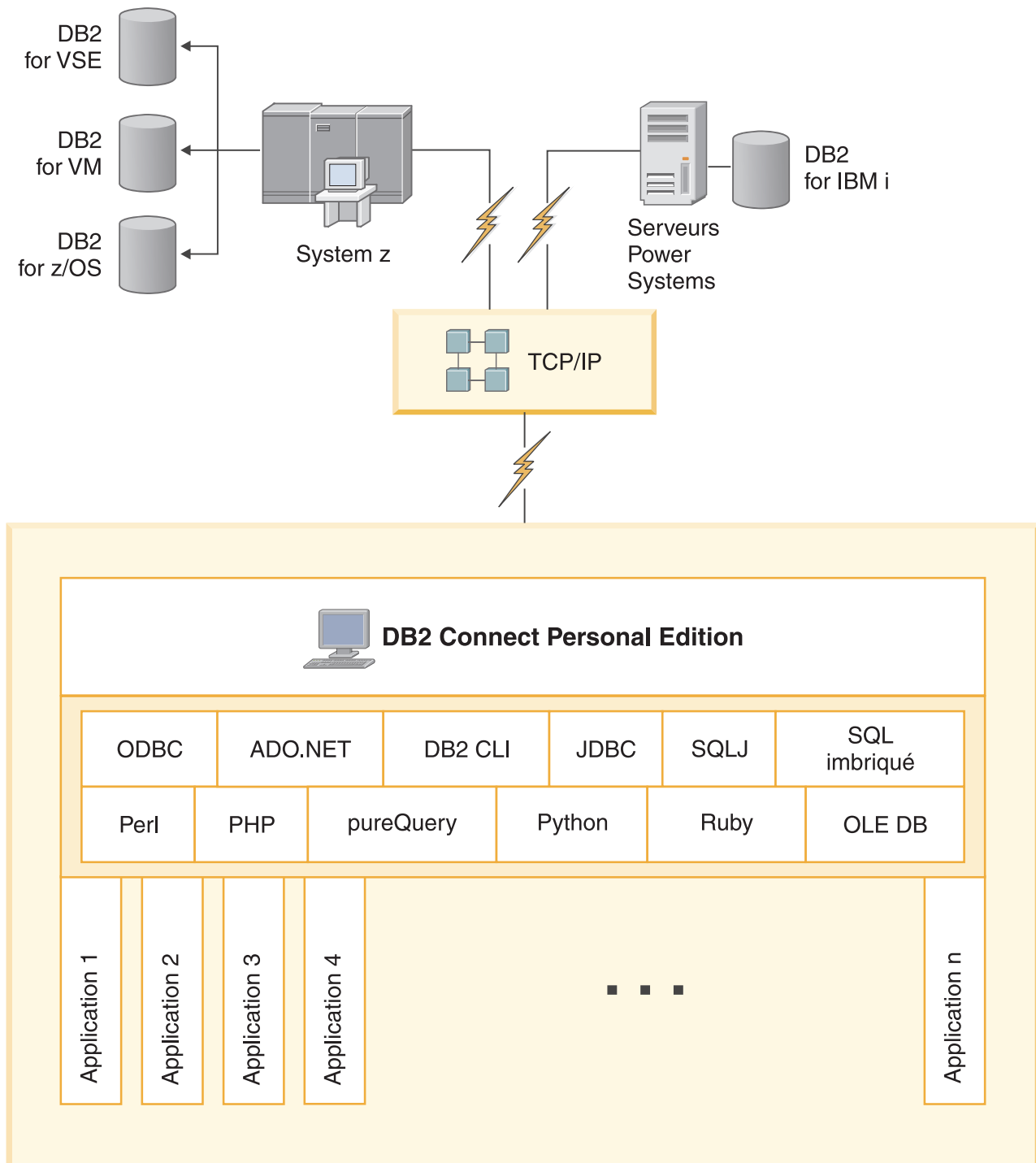


Figure 4. Connexion directe entre DB2 Connect et un serveur de base de données grand système IBM

Produits serveur DB2 Connect en tant que serveurs de connectivité

Un serveur DB2 Connect permet à plusieurs clients de se connecter aux données d'un grand système IBM et peut réduire significativement l'effort requis pour établir et gérer l'accès aux données d'entreprise. La figure 5 illustre la solution IBM pour les environnements dans lesquels vous souhaitez qu'un client DB2 puisse établir une connexion indirecte avec un serveur de base de données grand système IBM via un serveur DB2 Connect, tel que DB2 Connect Enterprise Edition.

Remarque : Les connexions indirectes sont prises en charge uniquement avec les clients DB2 ou JCC exécutés sous Linux, UNIX ou Windows. Une tentative de connexion à un serveur de base de données grand système IBM via un serveur DB2 Connect utilisant n'importe quel autre client renvoie une erreur SQL1334.

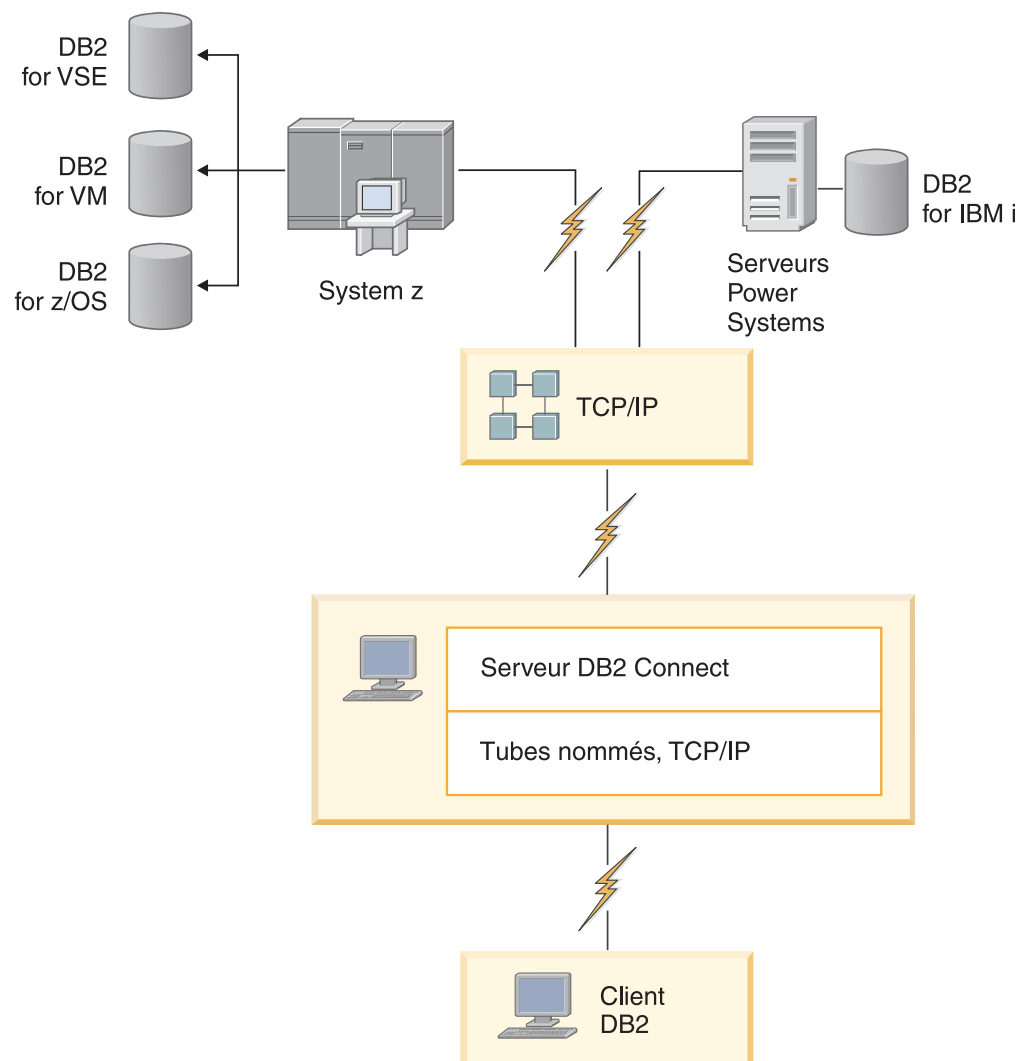


Figure 5. DB2 Connect Enterprise Edition

Si la connexion TCP/IP au serveur DB2 Connect est perdue, le client tentera automatiquement de rétablir la connexion. Le client tente tout d'abord de rétablir la connexion au serveur d'origine. S'il ne parvient pas à rétablir la connexion, le client bascule vers un serveur DB2 Connect de remplacement. (Ce serveur est spécifié sur l'instance du serveur et son emplacement est renvoyé au client au

cours de la connexion.) Si la connexion au serveur de remplacement n'est pas rétablie, le client tente de rétablir la connexion au serveur d'origine. Le client poursuit ses tentatives de rétablissement de la connexion, en passant du serveur d'origine au serveur de remplacement, jusqu'à ce que la connexion soit établie ou que le nombre de tentatives soit épuisé.

DB2 Connect et applications Web

Le navigateur Web est rapidement devenu l'interface standard de nombreux éléments, qu'il s'agisse de catalogues en ligne ou d'applications Intranet. Pour les applications Web simples, un seul serveur Web peut suffire. Pour les applications volumineuses qui requièrent un accès à la base de données et le traitement des transactions, IBM offre des solutions utilisant DB2 Connect pour gérer de grands nombres de transactions simultanées sur le Web.

Avantages et limitations de la programmation CGI traditionnelle

Les applications e-business du Web utilisent l'interface CGI (Common Gateway Interface) pour permettre aux utilisateurs d'interroger des bases de données d'arrière-plan. De nombreuses entreprises utilisent également des applications Web en interne qui possèdent généralement une base de données en arrière-plan.

Les utilisateurs remplissent des formulaires sur la page Web, formulaires qui sont soumis via l'interface CGI aux applications ou aux scripts sur le serveur Web. Le script utilise à son tour une API de base de données fournie pour soumettre des requêtes SQL à une base de données hôte. Le même script peut générer une page Web (HTML) qui est le résultat d'une requête et la renvoyer afin de l'afficher sur le navigateur Web de l'utilisateur. Un bon exemple serait un catalogue en ligne dans lequel l'utilisateur peut consulter la disponibilité des biens et des services ainsi que leurs prix actuels.

Les applications CGI peuvent être simples à concevoir et à gérer. Comme le standard CGI est libre de système d'exploitation et de langage, il est disponible sur pratiquement toutes les plateformes de programmation. Les programmes CGI peuvent être écrits en C++ ou dans un langage de scriptage, tel que Perl ou PHP.

Bien que l'interface CGI puisse sembler être la solution idéale pour les applications Web, elle possède cependant des inconvénients considérables. L'environnement de programmation de l'interface CGI n'est pas aussi sophistiqué que celui des autres API. De plus, l'évolutivité peut être un enjeu important dans le cadre d'opérations de commerce électronique de grande envergure. Chaque fois qu'une application CGI est invoquée, un nouveau processus est créé sur le serveur Web. Chaque processus doit établir sa propre connexion à la base de données et soumettre sa propre requête. Dans des environnements transactionnels volumineux, cette limitation peut engendrer des problèmes de performances importants.

Vous pouvez utiliser DB2 Connect avec un serveur Web pour créer des applications robustes de commerce électronique volumineuses. DB2 Connect offre plusieurs solutions qui améliorent les performances des applications Web. Les procédures mémorisées permettent aux utilisateurs de DB2 Connect de réduire le nombre de requêtes envoyées à la base de données.

Le regroupement de connexions réduit la fréquence des connexions à une base de données et des déconnexions d'une base de données.

Utilisation de PHP en tant que plug-in ou module de serveur Web

Bien que PHP puisse être utilisé pour la programmation CGI, il est généralement utilisé en tant que plug-in ou module de serveur Web. Dans un serveur Web incluant plusieurs processus (Apache, par exemple), le pilote IBM DB2 pour PHP peut permettre de limiter le problème d'évolutivité. Dans un processus Web incluant plusieurs processus, un pool de processus sont réutilisés afin de traiter les requêtes de serveur Web. Pour supprimer le besoin de création d'une connexion de base de données pour chaque requête Web, une connexion persistante peut être créée. Dans cet environnement, une connexion persistante peut exister au delà de la portée d'un seul script PHP. La connexion sera réutilisée si une connexion identique est requise par une requête Web suivante.

DB2 Connect et IBM WebSphere

IBM WebSphere offre une solution e-business plus complète que les solutions offertes par les outils de script traditionnels, tels que le langage PHP. WebSphere Application Server réalise les fonctions de script du langage PHP mais vous offre également des services complexes et de pointe à travers le Web, utilisant les servlets, les pages Active Server et Enterprise JavaBeans™ tout en prenant en charge les technologies Web telles que Java™, TCP/IP, HTTP, HTTPS, HTML, DHTML, XML, MIME, SMTP, IIOP et X.509. Avec WebSphere, vous pouvez :

- Exploiter les normes de l'industrie pour accélérer le développement et optimiser l'interopérabilité
- Brancher des outils et des cadres d'application tiers
- Analyser les performances et l'utilisation du contenu du site Web
- Régler votre site facilement afin de vous adapter à un nombre plus élevé d'utilisateurs et gérer le débit
- Effectuer des déploiements sur plusieurs environnements d'exploitation majeurs (AIX, HP-UX, Linux, Novell NetWare, z/OS, IBM i, système d'exploitation Solaris, Microsoft Windows)
- Utiliser votre serveur Web existant, notamment les serveurs Apache, IBM, Netscape et Microsoft.

WebSphere n'est pas un produit, mais une famille de trois produits qui s'adresse à trois marchés cible différents. Le coeur de la solution WebSphere est WebSphere Application Server.

WebSphere Application Server offre l'environnement pour trois types d'objet. Le premier type d'objet est le JSP (Java), qui sont des pages analogues aux pages Active Server. Le deuxième type d'objet est constitué de servlets Java et le troisième est Enterprise JavaBeans. Enterprise JavaBeans sont des normes émergentes pour le déploiement d'applications de classe d'entreprise robustes à très large échelle.

Les applications WebSphere peuvent être déployées sur la même plateforme que le serveur Web et que DB2. Dans le cas de DB2 for z/OS, DB2 Server for VM and VSE, DB2 for IBM i, WebSphere est déployé sur la même plateforme que le serveur DB2 Connect.

Il existe diverses solutions WebSphere et Rational Application Developer (RAD). Pour obtenir des informations détaillées, consultez la page Web <http://www.ibm.com/software/webservers/appserv/was/>

DB2 Connect en tant que serveur d'applications Java

De nombreux inconvénients associés aux langages de script peuvent être contournés en utilisant le langage Java. IBM fournit des applets et applications qui vous permettent de vous servir de Java à chaque étape d'une transaction Web. En mettant en oeuvre les solutions proposées par IBM, vous pouvez combiner différentes techniques et utiliser par exemple des solutions intégrant des scripts comme Perl DBI ou Microsoft Active Server Pages avec DB2, ou opter pour une implémentation plus robuste fournie par un serveur d'applications Java comme IBM WebSphere.

Les programmeurs Java disposent de deux interfaces API. La première, JDBC, prend en charge l'utilisation de Java pour développer des applets Java sensibles aux données, des applications Java, des servlets Java, des pages JSP (Java Server Pages) et des beans EJB (Enterprise Java Beans). JDBC est une API d'appel de méthode ou de niveau d'appel. La seconde interface API Java, SQLJ, vous permet de spécifier du code SQLJ incorporé dans un programme Java. DB2 peut utiliser les deux interfaces API, aussi bien du côté serveur que du côté client d'une transaction Web.

Du côté client, les applets, les applets sensibles aux données et les applications sont pris en charge. Du côté bases de données, l'activation Java prend la forme d'objets de base de données, comme des fonctions définies par l'utilisateur et des procédures mémorisées.

Pour DB2 for z/OS, DB2 Server for VM and VSE, et DB2 for IBM i, vous disposez de deux méthodes pour déployer une application Java. Vous pouvez utiliser la connectivité directe fournie par DB2 Connect Personal Edition via TCP/IP, ou vous pouvez utiliser un serveur DB2 Connect qui assurera la connectivité au serveur de base de données grand système IBM.

Dans les deux cas, l'utilisateur sur le Web n'a besoin d'aucun logiciel spécifique pour accéder à la base de données, il doit uniquement disposer d'un navigateur Web standard. Le seul composant qui doit être installé est le serveur DB2 Connect et tout serveur Web répondant aux normes de l'industrie. Si le serveur Web et DB2 Connect ne se trouvent pas sur les mêmes machines physiques, un client IBM Data Server doit être installé sur le serveur Web.

Pour DB2 for z/OS, le composant clé est constitué par un serveur DB2 Connect s'exécutant sur un serveur de niveau intermédiaire. Ce composant fournit l'activation du serveur JDBC, en plus de la connexion au serveurs DB2 for z/OS, DB2 Server for VM and VSE, et DB2 for i. Une fois encore, le client n'a pas besoin de posséder d'autre logiciel spécifique que le navigateur Web.

IBM propose une prise en charge étendue et des outils permettant de développer des applets et des applications Java. Pour le développement d'applications de base de données, DB2 Database Enterprise Developer Edition fournit Rational Web Developer, IBM Data Studio, DB2 WebSphere Application Server, ainsi que DB2 et DB2 Connect pour l'exécution de tests. Les outils tiers tels que NetBeans, Borland JBuilder ou Symantec Visual Cafe fonctionnent également avec les solutions de base de données IBM.

DB2 Connect sur le serveur Web

IBM fournit aux serveurs HTTP (Web) tous les produits DB2 Connect. Les serveurs DB2 Connect, tels que DB2 Connect Enterprise Edition, offrent une prise en charge prête à l'emploi pour les serveurs Web Apache ou Lotus Domino Go et peuvent fonctionner avec n'importe quel autre serveur Web tel que Microsoft Internet Information Server ou Netscape Enterprise Server.

Si vous travaillez avec la famille de bases de données DB2 sur des systèmes System z, IBM Power Systems, VM et VSE, un serveur DB2 Connect est requis sur le serveur Web. Les serveurs DB2 Connect fournissent les bibliothèques et les interfaces de communication permettant aux serveurs Web d'accéder à ces plateformes grand système IBM. Le protocole TCP/IP peut être utilisé pour les communications entre le serveur Web et une base de données opérant sous System z, IBM Power Systems, VM ou VSE.

Remarque : Les solutions Web IBM permettent d'utiliser plusieurs bases de données dans le même script CGI (Common Gateway Interface), tel que PHP, ou dans la même transaction d'un script CGI.

Procédures mémorisées

Un enjeu important pour les applications Web, ainsi que dans le monde client-serveur, est la réduction du trafic entre le serveur HTTP et la base de données d'arrière-plan. Cet enjeu est particulièrement important lors du traitement de transactions volumineuses, qui sont le cœur de la plupart des applications e-business.

L'approche recommandée consiste à combiner l'interface de programmation CGI avec la logique métier et la logique de programmation contenues dans les procédures mémorisées. DB2 Database for Linux, UNIX, and Windows, DB2 for z/OS, DB2 for IBM i, et DB2 partagent la même convention de paramètre pour l'appel de procédures mémorisées.

Tout comme avec les scripts d'interface ordinaires, le navigateur Web soumet le formulaire au serveur Web où le script d'interface Web est exécuté. Cependant, au lieu d'envoyer chaque instruction SQL individuelle à la base de données DB2, une requête d'exécution d'une procédure mémorisée est envoyée. Cette procédure mémorisée regroupe des instructions SQL qui auraient été envoyées individuellement. Les procédures mémorisées réduisent le nombre de messages circulant entre le script d'interface Web et la base de données d'arrière-plan.

L'avantage principal des procédures mémorisées est de réduire le trafic réseau entre le serveur HTTP et la base de données d'arrière-plan DB2.

DB2 Connect et serveurs d'applications

L'émergence des applications client-serveur a permis aux concepteurs d'améliorer la convivialité et de réduire les coûts de formation grâce au développement d'interfaces graphiques pour les applications sur des plateformes telles que Windows. Elle a également apporté la flexibilité de déléguer la fonction de gestion de base de données à des serveurs de base de données robustes sur divers systèmes d'exploitation et diverses plateformes logicielles.

Le modèle client-serveur, dans lequel la logique applicative est distribuée aux postes de travail client, est généralement désigné sous le terme de *serveur client à deux niveaux*. Dans un modèle à deux niveaux, l'application est déployée au niveau du client et le serveur de base de données implémente le serveur ou le niveau d'arrière-plan. DB2 Connect fournit une prise en charge complète des applications client-serveur à deux niveaux, où les serveurs de base de données sont DB2 for z/OS, DB2 for IBM i, ou DB2 Server for VM and VSE.

Avec l'augmentation de la taille des applications client-serveur, il est devenu évident que le modèle client-serveur à deux niveaux possède des limitations importantes. La distribution de grandes quantités de logiques applicatives à des centaines ou des milliers de postes de travail client a transformé la gestion des modifications en un procédé complexe et onéreux. Toute modification des règles métier requiert le remplacement de la partie client de l'application. Le déploiement de ces applications doit souvent avoir lieu simultanément sur tous les postes de travail client de l'entreprise afin de garantir une application efficace des règles métier.

Un autre inconvénient du modèle client-serveur à deux niveaux devenu évident avec la portée est la quantité de ressources consommées par ces applications. Le déploiement de centaines ou de milliers de *clients lourds*, tels que les clients à deux niveaux sont souvent appelés, augmente ainsi les demandes de puissance de traitement et de capacité de chaque poste de travail client. En outre, les demandes du serveur de base de données connaissent également une augmentation importante car chaque client requiert une connexion dédiée à la base de données et des ressources pour gérer cette connexion. Alors que la dépendance du client-serveur à deux niveaux en distribution de logiques applicatives peut être quelque peu réduite grâce à l'utilisation accrue de procédures mémorisées, d'autres inconvénients ne peuvent être traités sans modifier le modèle.

Solution pour un serveur d'applications

Comme le coût et la complexité des applications client-serveur à deux niveaux augmentent, la majeure partie des plus grandes applications ont décidé d'utiliser le client-serveur multiniveau. Dans le modèle multiniveau, le rôle du niveau de la base de données reste le même. Cependant, le niveau client est annexé d'un ou plusieurs niveaux intermédiaires, le plus souvent d'un niveau, d'où le nom à *trois niveaux*.

Dans le modèle à trois niveaux, le client est relégué à la gestion des interactions des utilisateurs et ne contient aucune logique applicative. La couche intermédiaire se compose d'un ou plusieurs serveurs d'applications. Le but du serveur d'applications est de fournir une implémentation robuste et rentable de la logique cachée derrière les processus et règles métier. Comme pour le modèle à deux niveaux, l'implémentation des règles métier est généralement renforcée par des procédures mémorisées visant à améliorer les performances.

Puisque les postes de travail client n'implémentent plus l'intégralité de la logique applicative et gèrent uniquement les interactions des utilisateurs, les besoins en ressources du niveau client ont été significativement réduits. Aussi, le niveau client du modèle à trois niveaux est souvent appelé *client léger*. En outre, comme un serveur d'applications centralisé se charge de la gestion des requêtes de tous les clients, il peut partager des ressources, telles que les connexions à la base de données de tous les clients. Ainsi, le serveur de base de données ne doit plus s'occuper de la gestion des connexions dédiées de chaque utilisateur de l'application.

De nombreux exemples d'applications à trois niveaux sont désormais mis en oeuvre dans le monde de l'industrie. Presque tous les fournisseurs de plannings ERP implémentent leurs applications à l'aide du modèle à trois niveaux, tels que les applications SAP R/3 et PeopleSoft V7. D'autres exemples englobent les fournisseurs d'ERM (Enterprise Relationship Management), tels que Siebel et Vantive.

Serveurs d'application et DB2 Connect

Les serveurs DB2 Connect offrent une prise en charge générale du déploiement des applications multiniveau. La prise en charge fournie par DB2 Connect englobe diverses API pouvant être utilisées pour développer la logique applicative (ODBC, ADO.NET, DB2 CLI, SQL imbriqué, JDBC, SQLJ, Perl, PHP et OLE DB), ainsi que comme infrastructure de communication complète pour l'interaction avec les serveurs de base de données de la famille DB2.

DB2 Connect prend également en charge les implémentations dans lesquelles un niveau de base de données se compose de plusieurs serveurs de base de données de la famille DB2. Ainsi, les applications peuvent implémenter des transactions qui mettent à jour des données résidant sur plusieurs serveurs de base de données au cours d'une même transaction.

Le protocole de validation à deux phases fourni par DB2 Connect assure l'intégrité de ces transactions réparties. Par exemple, une application peut, dans la même transaction, mettre à jour des données dans une base de données DB2 for z/OS et dans DB2 Database for Linux, UNIX, and Windows. Si la prise en charge des requêtes réparties est installée et activée, l'application peut lire une base de données Oracle et mettre à jour une base de données de la famille DB2 au cours de la même transaction.

Dans le diagramme suivant, les API ainsi que le mécanisme de connectivité entre le serveur d'applications et la base de données d'arrière-plan sont fournis par un serveur DB2 Connect tel que DB2 Connect Enterprise Edition.

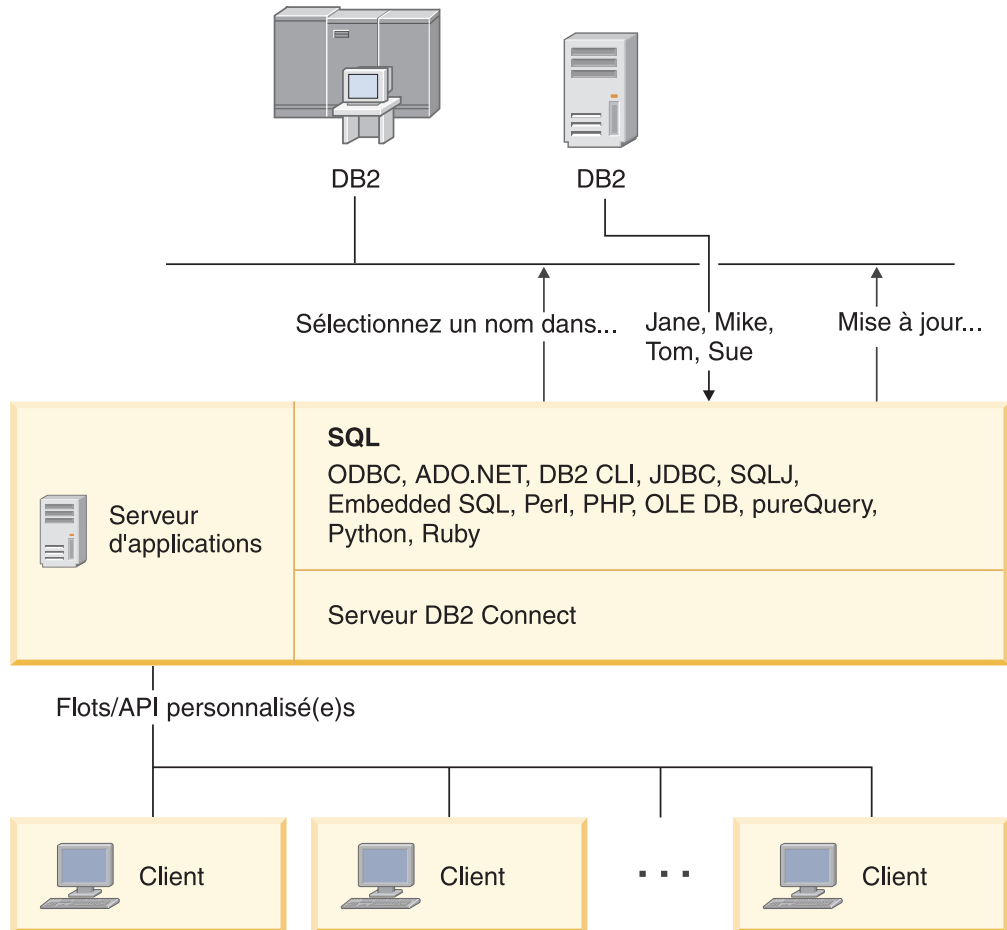


Figure 6. DB2 Connect prend en charge les serveurs d'applications

Des fonctions avancées de DB2 Connect, telles que les regroupements de connexions, peuvent réduire de manière significative les besoins en ressources des applications et simplifier l'implémentation des serveurs d'applications.

Configurations de DB2 Connect et des serveurs d'applications

Un serveur DB2 Connect est nécessaire pour pouvoir utiliser les serveurs d'applications. DB2 Connect Personal Edition n'est pas pris en charge et n'est pas sous licence pour l'utilisation des serveurs d'applications. En outre, les clients qui implémentent les serveurs d'applications doivent lire les conditions d'utilisation fournies avec leur copie de DB2 Connect pour comprendre le nombre de licences utilisateur qu'ils doivent acquérir.

Il existe deux méthodes pour déployer DB2 Connect dans un environnement de serveur d'applications. Un serveur DB2 Connect peut être installé sur :

- La machine du serveur d'applications
- Un serveur de communication distinct

Dans la plupart des cas, il est préférable d'installer une copie de DB2 Connect sur le même serveur que le serveur d'applications. L'installation de DB2 Connect sur le serveur d'applications lui permet de prendre part à tout schéma de reprise ou de répartition des charges mis en oeuvre par un serveur d'applications. Cette configuration peut également engendrer une amélioration des performances car elle supprime les tronçons de réseau supplémentaires requis lorsque DB2 Connect est installé sur un serveur distinct. L'administration est également simplifiée car aucun autre serveur supplémentaire ne doit être installé et géré.

L'installation de DB2 Connect sur un serveur distinct est une bonne option lorsque votre serveur DB2 Connect n'est pas disponible pour le système d'exploitation ou la plateforme logicielle sur lequel/laquelle le serveur d'applications fonctionne.

DB2 Connect et moniteurs de traitement de transactions

Un serveur d'applications permet à un grand nombre d'utilisateurs d'exécuter des applications tout en utilisant un minimum de ressources système. Un serveur d'applications peut être développé pour permettre l'appel de transactions coordonnées à partir d'applications exécutées par un serveur d'applications. Cette coordination des transactions est généralement connue sous le nom de moniteur de traitement de transactions (TP). Un moniteur TP fonctionne de concert avec un serveur d'applications.

Une *transaction* peut être envisagée comme un événement de routine, en règle générale, une demande de service, qui gère les opérations courantes d'une entreprise. Le traitement en bon ordre des transactions est le type de travail pour lequel les moniteurs TP ont été conçus.

Traitement des transactions

Chaque entreprise possède ses règles et ses procédures qui décrivent la manière dont elle est supposée fonctionner. Les applications utilisateur implémentant ces règles peuvent être appelées la *logique applicative*. Les transactions exécutées par ces applications métier sont souvent désignées sous les termes de traitement de transactions ou de traitement des transactions en ligne.

Les caractéristiques clés des traitements des transactions en ligne commerciaux sont les suivantes :

Utilisateurs nombreux

Il est fréquent qu'un traitement de transaction soit utilisé par la majorité du personnel d'une entreprise, car un nombre si élevé de personnes affecte l'état en cours des affaires.

Répétitif

La majeure partie des interactions avec l'ordinateur se résume souvent en l'exécution d'un même processus encore et encore. Par exemple, la saisie d'une commande ou le traitement des paiements sont des opérations utilisées tous les jours à maintes reprises.

Interactions brèves

La plupart des interactions que le personnel d'une entreprise a avec le système de traitement de transaction sont de courte durée.

Données partagées

Puisque les données représentent l'état de l'entreprise, il ne peut exister qu'une seule copie des données.

Intégrité des données

Les données doivent représenter l'état en cours de l'entreprise et doivent être cohérentes en interne. Par exemple, toute commande doit être associée à un enregistrement client.

Coût/transaction faible

Puisque le traitement des transactions représente un coût direct dans la pratique commerciale, le coût du système doit être moindre. DB2 Connect permet aux applications sous le contrôle d'un serveur d'applications opérant sous Linux, UNIX et Windows, d'exécuter des transactions sur le réseau LAN et les serveurs de base de données grand système IBM éloignés et de coordonner ces transactions à l'aide d'un moniteur TP.

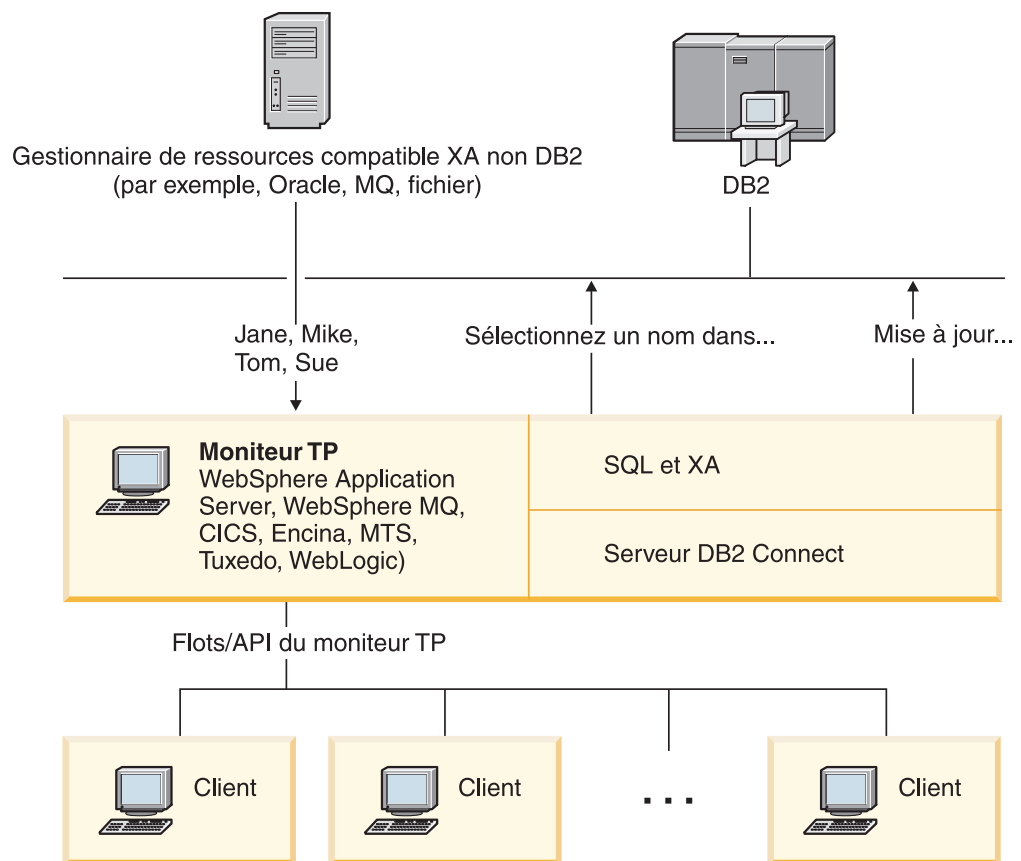


Figure 7. Prise en charge de DB2 Connect pour les moniteurs TP

Dans la figure 7, les API ainsi que le mécanisme de connectivité entre le serveur d'applications et la base de données d'arrière-plan sont fournis par un serveur DB2 Connect tel que DB2 Connect Enterprise Edition.

Exemples de moniteurs de traitement de transactions

Les moniteurs TP les plus courants sur le marché à l'heure actuelle sont :

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

Des serveurs de base de données IBM Power Systems, System z distants et des serveurs de base de données LAN peuvent être utilisés dans le cadre de transactions coordonnées par ces moniteurs TP.

Modèle DTP (Distributed Transaction Processing) X/Open

Une application exécutant la logique applicative peut être nécessaire pour mettre à jour diverses ressources au cours d'une même transaction. Par exemple, une application vide qui implémente un transfert d'argent d'un compte à un autre peut requérir le débit d'une base de données (le compte source) et le dépôt dans une autre base de données (le compte cible).

Il est également possible que divers fournisseurs vous procurent ces deux bases de données. Par exemple, l'un des bases de données pourrait être DB2 for z/OS et l'autre, une base de données Oracle. Au lieu que chaque moniteur TP implémente chaque interface de transaction propriétaire des fournisseurs de base de données, une interface de transaction commune entre un moniteur TP et n'importe quelle ressource à laquelle une application accède a été définie. Cette interface est connue sous le nom d'*interface XA*. Un moniteur TP qui utilise l'interface XA est désigné sous le terme de *gestionnaire de transactions (TM) compatible XA*. Une ressource actualisable qui implémente une interface XA est désignée sous le terme de *gestionnaire de ressources (RM) compatible XA*.

Les moniteurs TP répertoriés ci-dessus sont tous compatibles avec les gestionnaires de transactions (TM) compatibles XA. Les hôtes distants, les systèmes IBM Power Systems et les bases de données DB2 basées LAN, auxquels l'utilisateur accède via DB2 Connect, sont des gestionnaires de ressources compatibles XA. Par conséquent, tout moniteur TP disposant d'un gestionnaire de ressources compatible XA peut utiliser des bases de données hôte, IBM Power Systems, et DB2 basées LAN au sein d'applications métier exécutant des transactions.

Partie 2. Référence pour DB2 Connect

Chapitre 4. Mise à jour des répertoires de bases de données

DB2 Connect utilise les répertoires suivants pour gérer les informations relatives à la connexion à la base de données :

- Le *répertoire système des bases de données* qui contient le nom, le noeud et les informations d'authentification de chaque base de données à laquelle DB2 Connect accède.
- Le *répertoire des noeuds*, qui contient l'adresse réseau et les informations relatives au protocole de communication de chaque serveur de base de données grand système IBM auquel accède DB2 Connect.
- Le *répertoire des services de connexion de la base de données*, qui contient des informations spécifiques aux bases de données de serveur de base de données grand système IBM.

Remarque :

1. Avant de mettre à jour ces répertoires, vous devez configurer les communications sur le serveur de base de données grand système IBM et les postes de travail.
2. Les répertoires de base de données peuvent être mis à jour à l'aide de l'assistant de configuration (CA).

Pour mettre à jour les répertoires de base de données :

1. Collecte d'informations relatives au répertoire de base de données à l'aide du formulaire de personnalisation de répertoire
2. Voir la rubrique «Mise à jour des informations relatives aux serveurs de base de données éloignés dans les répertoires» dans le Centre de contrôle

Valeurs du répertoire système des bases de données

Il existe un répertoire système des bases de données pour chaque instance du gestionnaire de la base de données, qui contient une entrée pour chaque base de données cataloguée pour cette instance. Dans les produits DB2 Connect, le répertoire système des bases de données contient des informations sur le nom, l'alias, le nom de poste et le type d'authentification de chaque base de données.

Vous pouvez spécifier les informations suivantes dans le répertoire système des bases de données :

Nom de la base de données

La même valeur que celle saisie dans la table Paramètres du répertoire DCS.

Alias de base de données

Alias du serveur de base de données grand système IBM. Ce nom sera utilisé par un programme d'application qui accède à la base de données. Par défaut, la valeur que vous spécifiez en tant que Nom de la base de données est utilisée.

Format : 1 à 8 caractères alphanumériques à un octet, en ce compris le signe dièse (#), le a commercial (@), le symbole du dollar (\$) et le trait de soulignement (_). Il ne peut commencer par un trait de soulignement ou un nombre.

Nom de noeud

La même valeur que celle saisie dans la table Paramètres du répertoire des noeuds.

Authentification

Spécifie l'emplacement où la validation du nom d'utilisateur et du mot de passe aura lieu pour les connexions issues du serveur DB2 Connect. Les options valides sont : SERVER, SERVER_ENCRYPT, CLIENT, KERBEROS, SERVER_ENCRYPT_AES et DATA_ENCRYPT. Le type d'authentification GSSPLUGIN n'est pas pris en charge dans le répertoire système des bases de données.

Valeurs du répertoire des noeuds

Vous pouvez spécifier les informations suivantes dans le répertoire des noeuds :

Nom de noeud

Pseudonyme du serveur de base de données grand système IBM sur lequel réside la base de données distante. Ce nom est défini par l'utilisateur. Utilisez le même nom de noeud dans les tables Paramètres du répertoire des noeuds et Paramètres du répertoire système des bases de données.

Format : 1 à 8 caractères alphanumériques à un octet, en ce compris le signe dièse (#), le a commercial (@), le symbole du dollar (\$) et le trait de soulignement (_). Il ne peut commencer par un trait de soulignement ou un nombre.

Protocole

Il doit s'agir du protocole TCP/IP.

Type de sécurité

Le type de vérification de la sécurité qui sera effectué. Pour les noeuds TCP/IP, SECURITY SOCKS est une option qui spécifie le noeud qui sera sécurisé par socket, auquel cas les variables d'environnement SOCKS_NS et SOCKS_SERVER sont obligatoires et doivent être définies de sorte à activer la sécurisation par socket.

Nom d'hôte distant TCP/IP et adresse IP

Lorsque vous définissez le noeud TCP/IP, il s'agit du nom d'hôte éloigné TCP/IP ou de l'adresse IP éloignée. Si un nom d'hôte est spécifié, il doit être résolu dans le poste de travail DB2 Connect, via la recherche de serveur de noms de domaine (DNS) ou par une entrée dans le fichier hôte TCP/IP local.

Pour les hôtes DB2 for z/OS distants, le nom d'hôte figure dans le message DSNL004I message (DOMAIN=hostname) au lancement de l'utilitaire DDF (Distributed Data Facility). La commande -DISplay DDF peut également être utilisée.

Si vous accédez à un groupe de partage de données z/OS, le nom de domaine doit être mappé à l'adresse IP virtuelle du groupe dynamique DB2. Cette adresse se dirige vers le membre DB2 le moins chargé. Pour accéder à un membre spécifique, utilisez l'adresse IP virtuelle dynamique des membres DB2 et désactivez le routage sysplex. Chaque message DSNL004I du membre affiche le nom de domaine spécifique au membre.

Nom de service TCP/IP ou numéro de port

Lorsque vous définissez le noeud TCP/IP, il s'agit du nom de service TCP/IP éloigné ou du numéro de port. Il doit être défini sur TCP/IP au

niveau de l'hôte distant. Le numéro de port 446 a été enregistré en tant que numéro de port par défaut pour DRDA.

Pour les hôtes DB2 for z/OS distants, le numéro de port est défini dans l'ensemble de données d'amorce (BSDS) en tant que PORT et indiqué également dans le DSNL004I (TCPPORT=portnumber) au lancement de l'utilitaire DDF (Distributed Data Facility). La commande -DISplay DDF peut également être utilisée.

Si vous accédez à un groupe de partage de données z/OS, le nom de domaine doit être mappé à l'adresse IP virtuelle du groupe dynamique DB2. Cette adresse se dirige vers le membre DB2 le moins chargé. Pour accéder à un membre spécifique, utilisez l'adresse IP virtuelle dynamique des membres DB2 et désactivez le routage sysplex. Chaque message DSNL004I du membre affiche le nom de domaine spécifique au membre.

Remarque : Un second port utilisé pour les opérations de resynchronisation de validation en deux phases peut être attribué au serveur. Par exemple, l'ensemble de données d'amorce DB2 for z/OS attribue un numéro de port (RESPORT) à utiliser pour la resynchronisation des connexions entrantes dans DB2 for z/OS uniquement. Aucun nom de service ne doit être défini.

Valeurs du répertoire DCS

Vous pouvez spécifier les informations suivantes dans le répertoire DCS :

Nom de la base de données

Pseudonyme défini par l'utilisateur pour le serveur de base de données grand système IBM. Utilisez le même nom de base de données dans les tables Paramètres du répertoire DCS et Paramètres du répertoire système des bases de données.

Format : 1 à 8 caractères alphanumériques à un octet, y compris le signe dièse (#), le a commercial (@), le symbole du dollar (\$) et le trait de soulignement (_). Il ne peut commencer par un trait de soulignement ou un nombre.

Nom de la base de données cible

Base de données sur le système du serveur de base de données grand système IBM, comme suit :

System z

Un sous-système DB2 for z/OS identifié par son LOCATION NAME ou l'un des alias de noms LOCATION définis sur le serveur z/OS.

Le LOCATION NAME peut être déterminé lorsque vous vous connectez au TSO et que vous exécutez la requête SQL à l'aide de l'un des outils de requête disponibles :

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

Plusieurs LOCATION NAME sont également définis dans l'ensemble de données d'amorce (BSDS) ainsi que dans le message DSNL004I (LOCATION=location) qui apparaît au lancement de l'utilitaire DDF (Distributed Data Facility). La commande -DISplay DDF peut également être utilisée.

Si vous accédez à un groupe de partage de données z/OS, le nom de domaine doit être mappé à l'adresse IP virtuelle du groupe

dynamique DB2. Cette adresse se dirige vers le membre DB2 le moins chargé. Pour accéder à un membre spécifique, utilisez l'adresse IP virtuelle dynamique des membres DB2 et désactivez le routage sysplex. Chaque message DSNL004I du membre affiche le nom de domaine spécifique au membre.

VSE ou VM

Le nom de la base de données (DBNAME)

IBM Power Systems

Le nom de la base de données relationnelle (RDBNAME)

Divers

Pour les systèmes d'exploitation Windows, Linux et UNIX, l'alias de base de données trouvé dans le répertoire de base de données.

Chaîne de paramètres

Si vous souhaitez modifier les paramètres par défaut, spécifiez un ou tous les paramètres suivants en respectant l'ordre suivant.

map-file

Le nom du fichier de mappage SQLCODE qui remplace le mappage SQLCODE par défaut. Pour désactiver le mappage SQLCODE, spécifiez **NOMAP**.

Remarque : Lorsque vous traitez une demande de requête, le serveur DRDA renvoie des données sous la forme d'un ensemble de lignes représentant l'ensemble de résultats. A chaque ligne, une structure SQLCA est également renvoyée. Elle contient généralement un code sqlcode zéro ou positif (tel que +12 ou +802). Si vous utilisez un fichier de mappage personnalisé sur un serveur DB2 Connect, les codes sqlcode positifs ne seront pas mappés s'ils se trouvent dans le fichier de mappage personnalisé et s'ils possèdent des mappages personnalisés (par exemple, ils sont mappés vers un code sqlcode différent ou possède des mappages de jeton personnalisés).

Il est important de souligner que :

1. Les codes sqlcode positifs représentent des avertissements, par rapport aux codes sqlcode négatifs qui représentent des conditions d'erreur. Tous les codes sqlcode négatifs seront toujours mappés quoi qu'il advienne, indépendamment du fichier de mappage utilisé. Tous les codes sqlcode positifs, contenus dans un fichier de mappage personnalisé et mappés à eux-mêmes sans aucune modification, seront toujours mappés également. Aussi, les codes sqlcode positifs qui ne se trouvent pas dans le fichier de mappage personnalisé sur le serveur DB2 Connect seront également toujours mappés.
2. Si vous utilisez le fichier de mappage par défaut ou que vous vous connectez directement à la base de données hôte, le mappage du sqlcode sera toujours effectué pour tous les codes sqlcode.

,D Il s'agit du second paramètre à position fixe. S'il est spécifié, l'application se déconnectera de la base de

données du serveur de base de données grand système IBM lorsque l'un des codes SQLCODES suivants est renvoyé :

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

Lorsque le paramètre de déconnexion ,D n'est pas spécifié, la déconnexion sera effectuée uniquement lorsque les codes SQLCODE suivants sont renvoyés :

SQL30020N
SQL30021N
SQL30041N
SQL30061N
SQL30081N

Pour obtenir des explications sur ces codes, reportez-vous à la rubrique *Message Reference*.

Remarque : Si DB2 Connect se déconnecte suite à une erreur, une annulation sera effectuée automatiquement.

„INTERRUPT_ENABLED

Il s'agit du troisième paramètre à position fixe. INTERRUPT_ENABLED s'applique uniquement si le serveur de fin ne prend pas en charge les interruptions. Si un serveur prend en charge le flot d'interruption DRDA, DB2 Connect transmet simplement la demande d'interruption au serveur.

Si INTERRUPT_ENABLED a été configuré dans le répertoire DCS sur le poste de travail DB2 Connect et qu'une application cliente émet une interruption alors qu'elle est connectée au serveur de base de données grand système IBM, DB2 Connect réalise l'interruption en supprimant la connexion et en ramenant l'unité d'oeuvre à son état antérieur. Ce comportement d'interruption est pris en charge sous AIX et Windows.

L'application reçoit le code sqlcode (-30081) indiquant que la connexion au serveur a pris fin. L'application doit alors établir une nouvelle connexion avec le serveur de base de données grand système IBM afin de traiter des requêtes de base de données supplémentaires. Sur des plateformes autres que AIX version 5.2 et version ultérieure et Windows, DB2 Connect ne prend pas en charge l'option de déconnexion automatique lorsqu'une application qui l'utilise reçoit une demande d'interruption.

Remarque : Cette prise en charge fonctionne pour les protocoles TCP/IP sur n'importe quelle plateforme. Le

client peut arrêter le socket, mais, en fonction de la mise en oeuvre du serveur, il peut ou non y avoir une demande en attente. DB2 for z/OS utilise les appels de socket asynchrones et est par conséquent capable de détecter la perte de connexion et d'annuler toute instruction SQL de longue durée en cours d'exécution.

,,,,,SYSPLEX

Ce paramètre (sixième paramètre à position fixe) peut être utilisé pour activer de manière explicite la prise en charge DB2 Connect SYSPLEX pour une base de données particulière.

,,,,,LOCALDATE=<valeur>

Ce paramètre (le septième paramètre à position fixe) est utilisé pour activer la prise en charge du format de date DB2 Connect. Il est implémenté à l'aide d'un masque de date de la <valeur>, comme suit :

Supposons que vous exécutez les instructions de l'interpréteur de commandes (CLP) suivantes :

```
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
```

L'alias de base de données *newyork1* doit être utilisé pour accéder à la base de données hôte sans modifier les dates car aucun masque de date n'a été spécifié.

Cependant, avec le nouveau support de format de date, vous pouvez désormais utiliser les commandes suivantes de l'interpréteur de commandes. Dans ce cas, puisque l'interpréteur de commandes est utilisé et que la chaîne de paramètres est spécifiée à l'aide de guillemets, la valeur de LOCALDATE doit être spécifiée entre deux paires de guillemets. Notez que l'utilisation du caractère d'échappement du système d'exploitation "\\" (barre oblique inversée) garantit que les guillemets ne seront pas enlevées de la spécification LOCALDATE.

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

L'alias de base de données *newyork2* vous donne accès à la même base de données hôte et dispose, en outre, d'un masque de format de date spécifié. Cet exemple illustre le masque de format de date spécifié à l'aide du mot clé LOCALDATE et du septième paramètre à position fixe dans le champ PARMS d'une entrée de répertoire DCS.

Pour que le masque de date soit valide, TOUTES les conditions suivantes doivent être vraies :

1. Il ne peut y avoir tout au plus qu'une seule séquence de Y, M et D où Y représente le numéro d'une année, M le numéro d'un mois et D le numéro d'un jour.
2. Le nombre maximal de Y dans une séquence est de quatre.

3. Le nombre maximal de M dans une séquence est de deux.
4. Le nombre maximal de D dans une séquence est de deux.

Les exemples suivants sont des exemples de masque de date valides :

- "YYyyMmDd" - les chiffres de Y, M, et D sont insensibles à la casse
- "MM+DD+YYYY" - vous pouvez avoir un masque de plus de 10 octets et posséder des caractères autres que Y, M et D dans le masque
- "abcYY+MM" - Vous pouvez ne pas avoir de séquence de D

Les exemples suivants sont des exemples de masque de date non valides :

- "YYYYyMMDD" - non valide car 5 Y sont présents dans la séquence
- "YYYYMDDM" - invalide car deux séquences de M sont présentes

Si un masque de format de date n'est pas valide, aucune erreur ne sera émise. Il sera ignoré. La validité d'un masque de date ne signifie pas qu'il sera utilisé. La conversion du format de date en un masque de date valide sera uniquement effectuée si TOUTES les conditions suivantes sont vraies :

1. Il n'y a pas d'erreur SQL.
2. Le résultat est une valeur de date dans un format de style ISO (ISO et JIS)
3. La zone de données de sortie possède une longueur minimale de 10 octets. Il s'agit de la taille minimale d'une zone de données de sortie suffisante pour y conserver une valeur de données même si AUCUNE modification du format de date ne doit avoir lieu. Cette exigence s'applique même si le masque de format de date a une longueur inférieure à 10 octets.
4. Un masque de format de date valide est spécifié dans l'entrée de répertoire DCS dont la taille convient à la zone de données de sortie.

,,,,,,BIDI=<ccsid>

Ce paramètre (le neuvième paramètre à position fixe) est utilisé pour spécifier le CCSID bidirectionnel (BiDi) à utiliser pour remplacer le CCSID BiDi par défaut du serveur de base de données. Par exemple :

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

où *xyz* représente le nouveau CCSID.

Feuille de travail de personnalisation du répertoire

La feuille de travail de personnalisation du répertoire indique les informations que vous devez rassembler. Il se peut que vous préféreriez effectuer une copie de cette feuille de travail et saisir les valeurs système.

Paramètres du répertoire des noeuds

Tableau 1. Paramètres du répertoire des noeuds

Paramètre	Exemple	Votre valeur
Nom de noeud	DB2NODE	
Nom d'hôte distant (noeud TCP/IP)	ZOSHOST	
Serveur (nom de service TCP/IP ou numéro de port)	db2inst1c (ou 446)	

Remarque :

1. Le numéro de port TCP/IP par défaut pour DRDA est 446
2. Sauf si vous savez que le serveur de base de données grand système IBM prend en charge SECURITY SOCKS, ne spécifiez pas SECURITY pour un noeud TCP/IP.

Paramètres du répertoire DCS

Tableau 2. Paramètres du répertoire DCS

Paramètre	Exemple	Votre valeur
Nom de la base de données	DB2DB	
Nom de la base de données cible	NEW_YORK3	
Demandeur d'application		
Chaîne de paramètres	" ,,,,,LOCALDATE=\\\"YMMDD\\\"\\\""	

Paramètres du répertoire système des bases de données

Tableau 3. Paramètres du répertoire système des bases de données

Paramètre	Exemple	Votre valeur
Nom de la base de données	DB2DB	
Alias de base de données	NYC3	
Nom de noeud	DB2NODE	
Authentification	SERVER	

Définition d'entrées multiples pour la même base de données

Pour chaque base de données, vous devez définir au moins une entrée dans chacun des trois répertoires (le répertoire de noeuds, le répertoire DCS et le répertoire système des bases de données). Dans certains cas, vous souhaitez peut-être définir une entrée pour la base de données.

Par exemple, vous souhaitez peut-être désactiver le mappage SQLCODE pour les applications portées depuis le serveur de base de données grand système IBM tout en acceptant le mappage par défaut pour celles développées pour l'environnement client/serveur. Procédez alors comme suit :

- Définissez une entrée dans le répertoire des noeuds.
- Définissez deux entrées dans le répertoire DCS, avec des noms de base de données différents. Pour l'une des entrées, spécifiez NOMAP dans la chaîne de paramètres.

- Définissez deux entrées dans le répertoire système des bases de données, avec des alias de base de données différents et les deux noms de base de données spécifiés dans le répertoire DCS.

Les deux alias accèdent à la même base de données, l'un avec le mappage SQLCODE et l'autre sans mappage SQLCODE.

Traitement des données bidirectionnelles (BiDi)

La section suivante concerne uniquement les serveurs z/OS. Cette fonction ne doit pas être activée pour un serveur DB2 for IBM i étant donné que celui-ci assure déjà la prise en charge complète des données bidirectionnelles.

Les attributs BiDi sont requis pour la bonne gestion des données BiDi sur diverses plateformes :

- Format de numérotation (ARABIC versus HINDI)
- Orientation (RIGHT-TO-LEFT versus LEFT-TO-RIGHT)
- Mise en forme (SHAPED versus UNSHAPED)
- Permutation symétrique (YES ou NO)
- Type de texte (LOGICAL versus VISUAL)

Puisque les valeurs par défaut définies dans les diverses plateformes ne sont pas les mêmes, des incidents se produisent lors de l'envoi de données DB2 d'une plateforme à une autre. Par exemple, les plateformes Windows utilisent des données LOGICAL UNSHAPED alors que les données z/OS sont généralement au format SHAPED VISUAL. Par conséquent, en l'absence d'une prise en charge des attributs BiDi, les données envoyées depuis DB2 for z/OS à DB2 Connect sous Windows ne s'affichent pas correctement.

Lors de l'échange de données entre DB2 Connect et une base de données et un serveur, le récepteur effectue généralement la conversion des données entrantes. La même convention devrait normalement s'appliquer à la transformation de l'affichage des données BiDi, qui est un procédé supplémentaire au procédé de conversion des pages de codes habituel. Cependant, à l'heure actuelle, aucune hôte DB2 ne prend en charge les CCSID BiDi ou la transformation de l'affichage des données BiDi. Aussi, DB2 Connect a été amélioré et possède la fonction optionnelle de procéder à la transformation de l'affichage des données BiDi pour les données qui vont être envoyées à la base de données du serveur en plus de procéder à la transformation des données reçues de la base de données du serveur.

Pour que DB2 Connect procède à la transformation de l'affichage des données BiDi pour les données sortantes vers une base de données du serveur, le CCSID BiDi de la base de données du serveur devra être supprimé. Cette suppression est réalisée via l'utilisation du paramètre BIDI dans le champ PARMs de l'entrée de répertoire de base de données DCS du serveur de base de données.

Le procédé d'utilisation de cette fonction s'explique plus facilement au moyen d'un exemple.

Considérons un client IBM Data Server hébreu exécutant le CCSID 62213 (chaîne BiDi de type 5) qui souhaite accéder à une base de données hôte DB2 exécutant le CCSID 424 (chaîne BiDi de type 4). Cependant, vous savez que les données contenues dans la base de données hôte DB2 sont basées sur le CCSID 62245 (chaîne BiDi de 10).

Ce cas de figure engendre deux problèmes. Le premier problème est que la base de données hôte DB2 ne connaît pas la différence entre les types de chaîne BiDi possédant les CCSID 424 et 62245. Le second problème est que la base de données hôte DB2 ne reconnaît pas le CCSID 62213 du client IBM Data Server. Il prend uniquement en charge le CCSID 62209 (chaîne BiDi de type 10) basé sur la même page de codes que le CCSID 62213.

Vous devez vérifier que les données envoyées à la base de données hôte DB2 possèdent le format de type de chaîne BiDi 6 pour commencer et informer DB2 Connect qu'il doit procéder à la transformation de l'affichage BiDi des données qu'il reçoit de la base de données hôte DB2. Vous utiliserez le catalogage suivant pour la base de données hôte DB2 :

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

Cette instruction indique à DB2 Connect de remplacer le CCSID de la base de données hôte DB2 424 par 62245. Ce remplacement inclut la procédure suivante :

1. DB2 Connect se connecte à la base de données hôte DB2 à l'aide du CCSID 62209 (chaîne BiDi de type 10).
2. DB2 Connect procède à la transformation de l'affichage BiDi des données qu'il va envoyer à la base de données hôte DB2, du CCSID 62213 (chaîne BiDi de type 5) vers le CCSID 62209 (chaîne BiDi de type 10).
3. DB2 Connect procède à la transformation de l'affichage BiDi des données qu'il reçoit de la base de données hôte DB2, du CCSID 62245 (chaîne BiDi de type 10) au CCSID 62213 (chaîne BiDi de type 5).

Remarque :

1. La variable d'environnement ou la valeur de registre **DB2BIDI** doit être définie sur YES pour que le paramètre BIDI prenne effet. **DB2BIDI** doit être défini sur le poste de travail DB2 Connect sur lequel l'entrée de répertoire de base de données DCS est cataloguée. Pour les applications s'exécutant sur un client d'un serveur DB2 Connect éloigné, la variable **DB2BIDI** doit également être définie sur ce client.
2. Si vous souhaitez que DB2 Connect effectue la transformation de l'affichage des données qu'il va envoyer à la base de données hôte DB2 sans remplacer son CCSID, vous devez ajouter le paramètre BIDI dans le champs PARMS du répertoire de base de données DCS. Dans ce cas, le CCSID que vous devez fournir sera le CCSID par défaut de la base de données hôte DB2.
3. Dans certains cas de figure, l'utilisation d'un CCSID bidirectionnel peut entraîner la modification de la requête SQL au point qu'elle ne soit plus reconnue par le serveur DB2. Essayez d'éviter d'utiliser les CCSID IMPLICIT CONTEXTUAL et IMPLICIT RIGHT-TO-LEFT lorsque vous pouvez utiliser un autre type de chaîne. Les CCSID CONTEXTUAL peuvent donner lieu à des résultats imprévisibles si la requête SQL contient des chaînes de caractères délimitées. N'utilisez pas de chaînes de caractères délimitées dans les instructions SQL. Utilisez autant que possible des variables hôte.

Si un CCSID bidirectionnel déterminé cause des problèmes qui ne peuvent être résolus à l'aide des recommandations susmentionnées, définissez la variable d'environnement ou la valeur de registre **DB2BIDI** sur NO.

Spécifications des chaînes de paramètres

Les exemples suivants illustrent des paramètres DCS (chaque ligne représente un ensemble de paramètres) :

```
NOMAP
/u/username/sql1lib/map/dcs1new.map,D
,D
,,INTERRUPT_ENABLED
NOMAP,D,INTERRUPT_ENABLED,,,SYSPLEX,LOCALDATE="YYMMDD",,
```

Vous pouvez également accepter les valeurs par défaut ; pour ce faire, ne spécifiez pas la chaîne de paramètres.

Remarque : Utilisez le caractère d'échappement du système d'exploitation "\" (barre oblique inversée) lorsque vous utilisez l'interpréteur de commandes à partir de la ligne de commande du système d'exploitation sur les systèmes UNIX, car vous devez entrer deux paires de guillemets lorsque vous spécifiez le masque LOCALDATE dans la chaîne de paramètres. Par exemple :

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

Vous obtiendrez l'entrée de répertoire DCS suivante :

Entrée DCS 1 :

```
Nom de la base de données locale = X
Nom de la base de données cible = Y
Nom du demandeur d'application =
Paramètres DCS = ,,,,,,LOCALDATE="YYMMDD"
Commentaire =
Niveau d'édition du répertoire DCS = 0x0100
```

Chapitre 5. Sécurité de DB2 Connect

L'authentification des utilisateurs est importante avec DB2 Connect car les utilisateurs peuvent avoir un accès local ou éloigné à DB2 Connect et à la base de données contenant les données qu'ils souhaitent consulter. Les connexions accréditées et le support de Kerberos sont présentés ici ainsi que diverses considérations de sécurité pour les bases de données sur les machines hôte.

Connexions sécurisées via DB2 Connect

Certains serveurs de base de données DB2 prennent en charge les contextes sécurisés. Un *contexte sécurisé* permet à un administrateur de base de données de définir les conditions suivant lesquelles une application client pourra créer une connexion sécurisée. Une *connexion sécurisée* est autorisée à effectuer des actions qu'il est impossible de réaliser avec une connexion normale.

Il existe deux types de connexion sécurisée, implicite et explicite. Lorsque vous créez une connexion, vous obtenez une connexion sécurisée explicite, une connexion sécurisée implicite ou une connexion ordinaire. La connexion obtenue est régie par le fait que vous ayez ou non demandé une connexion sécurisée et par le fait que la connexion répond aux critères définis dans le contexte sécurisé sur le serveur, comme il est présenté dans le tableau 4.

Tableau 4. Quel type de connexion est généré suite aux différentes combinaisons d'action ?

	La connexion respecte les critères du serveur pour être sécurisée	La connexion ne respecte pas les critères du serveur pour être sécurisée
Vous avez demandé que la connexion soit sécurisée	Connexion sécurisée explicite	Une connexion ordinaire et l'avertissement SQL20360W (SQLSTATE 01679) est renvoyé.
Vous n'avez pas demandé que la connexion soit sécurisée	Connexion sécurisée implicite	Connexion ordinaire

Une *connexion sécurisée implicite* est identique à une connexion ordinaire, sauf qu'elle accorde des privilèges temporaires à l'utilisateur tant qu'il utilise la connexion. Les privilèges accordés sont définis dans le contexte sécurisé à l'origine de la sécurisation de la connexion.

Des connexions sécurisées implicites peuvent être créées par toute application qui se connecte à l'aide de DB2 Connect. Les connexions sécurisées implicites sont créées et utilisées de la même manière que des connexions ordinaires. Ceci signifie qu'aucune modification de code n'est nécessaire pour qu'une application existante puisse exploiter des connexions sécurisées implicites pour autant qu'elle se connecte via DB2 Connect.

Une *connexion sécurisée explicite* accorde des privilèges temporaires à l'utilisateur de la même manière qu'une connexion sécurisée implicite. Une connexion sécurisée explicite vous permet également de changer l'ID autorisation utilisé lors de l'exécution d'actions dans cette connexion. Le changement d'ID autorisation sur une connexion sécurisée explicite est appelé *changement d'utilisateurs*. Les ID

autorisation que vous pouvez utiliser et le fait qu'un mot de passe soit requis pour un ID autorisation donné sont définis comme partie du contexte sécurisé qui a permis la création de la connexion sécurisée.

Le changement d'utilisateur peut considérablement réduire le temps requis pour le partage d'une connexion entre plusieurs utilisateurs, plus particulièrement lorsqu'aucun mot de passe n'est requis car dans ce cas, le serveur de base de données n'authentifie pas l'ID autorisation. Toutefois, lors de l'utilisation de la fonction, vous devez être certain que votre application ne permet pas de changer d'ID autorisation sans valider et authentifier l'ID autorisation. Sinon, vous créez une brèche de sécurité dans votre système.

Des connexions sécurisées explicites peuvent être créées et il est possible de changer d'utilisateur lors de la connexion via DB2 Connect à l'aide de CLI ou de JDBC, y-compris pour les connexions établies par XA. La création d'une connexion sécurisée explicite et le changement d'utilisateur requièrent la définition d'attributs de connexion spéciaux. Cela signifie que les applications existantes doivent être modifiées afin que vous puissiez tirer le meilleur parti des connexions sécurisées explicites.

Outre les différences mentionnées précédemment, vous pouvez utiliser une connexion sécurisée (qu'elle soit implicite ou explicite) de la même manière que vous utiliseriez une connexion ordinaire. Toutefois, vous devez déconnecter expressément une connexion sécurisée explicite une fois que vous avez fini de l'utiliser même si elle est déconnectée. Sinon, les ressources utilisées par la connexion pourraient ne pas être libérées. Ce problème ne survient pas avec les connexions sécurisées implicites.

Remarque :

1. **Important :** Le fait de changer d'utilisateur sans indiquer de mot de passe supprime l'étape d'authentification du serveur de base de données. Votre application ne doit pas autoriser un changement d'ID autorisation sans mot de passe sauf si votre application a déjà validé et authentifié l'ID autorisation. Sinon, une brèche de sécurité est créée.
2. Les connexions sécurisées explicites ne doivent pas utiliser l'authentification CLIENT. Cette remarque ne s'applique pas aux connexions sécurisées implicites.
3. Les applications utilisant des connexions sécurisées explicites doivent être exécutées sur des machines sécurisées protégées par mot de passe et accessibles uniquement au personnel autorisé. Cette remarque ne s'applique pas aux connexions sécurisées implicites.

Création et arrêt d'une connexion sécurisée à l'aide de CLI

Si le serveur de base de données auquel vous vous connectez est configuré afin de permettre cette action, vous pouvez créer une connexion sécurisée explicite lors de la connexion via CLI.

Cette procédure suppose que vous n'utilisez pas de gestionnaire de transactions XA. Si vous utilisez un gestionnaire de transactions XA, il vous suffit de vous assurer que le gestionnaire de transactions est configuré de telle sorte que la valeur de configuration TCTX soit true lorsqu'il appelle xa_open. Si cette vérification a été effectuée, toute connexion pouvant être une connexion sécurisée explicite le sera. Pour vérifier qu'une connexion est une connexion sécurisée explicite, voir l'étape 3.

- La base de données à laquelle vous vous connectez doit prendre en charge les contextes sécurisés.
- Un contexte sécurisé doit être défini afin de reconnaître le client en tant que client fiable.
- Vous devez connaître l’ID autorisation système indiqué dans le contexte sécurisé. L’ID autorisation système d’une connexion sécurisée est l’ID autorisation fourni au serveur en tant que nom d’utilisateur lors de la création de la connexion. Pour que votre connexion soit sécurisée par un contexte sécurisé particulier, l’ID autorisation système doit être celui indiqué dans le contexte sécurisé. Demandez à l’administrateur système un ID autorisation système valide et le mot de passe correspondant à cet ID.

Les exemples illustrés dans les instructions suivantes sont rédigés en langage C et supposent que `conn` est un pointeur vers un descripteur de connexion valide non connecté. La variable `rc` est supposée posséder le type `SQLRETURN`.

1. Outre la configuration d’attributs de connexion qui doivent être définis pour une connexion régulière, définissez l’attribut de connexion `SQL_ATTR_USE_TRUSTED_CONTEXT` sur `SQL_TRUE` avec un appel de la fonction `SQLSetConnectAttr`.

```
rc = SQLSetConnectAttr(
    conn,
    SQL_ATTR_USE_TRUSTED_CONTEXT, SQL_TRUE, SQL_IS_INTEGER
);
```
2. Connectez-vous à la base de données comme vous le feriez pour une connexion ordinaire en appelant la fonction `SQLConnect`, par exemple. Utilisez l’ID autorisation système en tant que nom d’utilisateur et son mot de passe en tant que mot de passe. Veillez à vérifier les erreurs et avertissements, notamment celles et ceux répertoriés dans le tableau 5.

Tableau 5. Erreur indiquant l’échec de la création d’une connexion sécurisée

SQLCODE	SQLSTATE	Signification
SQL20360W	01679	La connexion n’a pu être établie en tant que connexion sécurisée. Elle a été établie en tant que connexion régulière.

Si aucune erreur ou aucun avertissement ne vous indique le contraire, une connexion sécurisée explicite est établie.

3. (Facultatif) Vous pouvez vérifier qu’une connexion établie est une connexion sécurisée explicite en vérifiant la valeur de l’attribut de connexion `SQL_ATTR_USE_TRUSTED_CONTEXT` à l’aide de la fonction `SQLGetConnectAttr`. S’il a la valeur `SQL_TRUE`, la connexion est une connexion sécurisée explicite.
4. Une fois que vous avez fini d’utiliser la connexion, vous devez la déconnecter explicitement même si elle est déconnectée. Si vous ne déconnectez pas de manière explicite une connexion sécurisée explicite, certaines ressources utilisées par la connexion peuvent ne pas être libérées.

Remarque :

1. Les connexions sécurisées explicites ne doivent pas utiliser l’authentification `CLIENT`. Cette remarque ne s’applique pas aux connexions sécurisées implicites.
2. Les applications utilisant des connexions sécurisées explicites doivent être exécutées uniquement sur des ordinateurs sécurisés protégés par mot de passe et accessibles uniquement au personnel autorisé. Cette remarque ne s’applique pas aux connexions sécurisées implicites.

Changement d'utilisateurs sur une connexions sécurisée via CLI

Vous pouvez changer d'utilisateur sur une connexion sécurisée explicite via l'interface de ligne de commande (CLI). Pour obtenir une définition du concept de changement d'utilisateur, voir la rubrique dans les liens connexes.

- La connexion doit avoir été créée en tant que connexion sécurisée explicite.
- La connexion sécurisée explicite ne doit pas être dans une transaction.
- Le contexte sécurisé qui a permis la création de la connexion sécurisée explicite doit être configuré afin que vous puissiez utiliser l'ID autorisation souhaité.

Les exemples illustrés dans les instructions suivantes sont rédigés en langage C et supposent que `conn` est un pointeur vers une connexion sécurisée explicite. La variable `rc` est supposée posséder le type `SQLRETURN`. Il est supposé que la variable `newuser` est un pointeur vers une chaîne de caractères contenant l'ID autorisation de l'utilisateur que vous souhaitez utiliser. Il est supposé que la variable `passwd` est un pointeur vers une chaîne de caractères contenant le mot de passe de cet ID autorisation.

1. Appelez la fonction `SQLSetConnectAttr` afin de définir l'attribut `SQL_ATTR_TRUSTED_CONTEXT_USERID`. Attribuez-lui l'ID autorisation que vous souhaitez utiliser.

```
rc = SQLSetConnectAttr(  
    conn,  
    SQL_ATTR_TRUSTED_CONTEXT_USERID, newuser, SQL_NTS  
);  
//Check for errors
```

Veillez à vérifier les erreurs et avertissements, notamment celles et ceux répertoriés dans le tableau 6.

Tableau 6. Erreurs indiquant l'échec de définition d'un nouvel ID autorisation lors du changement d'utilisateur

SQLCODE	Signification
CLI0106E	La connexion n'est pas connectée.
CLI0197E	La connexion n'est pas une connexion sécurisée.
CLI0124E	La valeur fournie n'est pas correcte. Vérifiez qu'elle n'est pas de type null, ou qu'elle n'est trop longue, par exemple.
CLI0196E	La connexion est impliquée dans une unité de travail qui l'empêche de changer d'utilisateur. Pour pouvoir changer d'utilisateur, la connexion ne doit pas se trouver dans une transaction.

2. (Facultatif sauf si le contexte sécurisé qui a permis cette connexion sécurisée requiert un mot de passe pour l'ID autorisation que vous souhaitez utiliser). Appelez la fonction `SQLSetConnectAttr` afin de définir l'attribut `SQL_ATTR_TRUSTED_CONTEXT_PASSWORD`. Attribuez-lui le mot de passe du nouvel ID autorisation.

```
rc = SQLSetConnectAttr(  
    conn,  
    SQL_ATTR_TRUSTED_CONTEXT_PASSWORD, passwd, SQL_NTS  
);  
//Check for errors
```

Veillez à vérifier les erreurs et les avertissements qui sont répertoriés dans le tableau 6 et le tableau 7, à la page 49.

Tableau 7. Erreurs indiquant l'échec de définition d'un nouvel mot de passe lors du changement d'utilisateur

SQLCODE	Signification
CLI0198E	L'attribut SQL_ATTR_TRUSTED_CONTEXT_USERID n'a pas encore été défini.

- Procédez comme avec une connexion ordinaire. Si vous utilisez un gestionnaire de transactions XA, le changement d'utilisateur est tenté lors de la requête suivante. Sinon, cette tentative est effectuée juste avant le lancement de l'appel de fonction suivant qui accède à la base de données (SQLExecDirect, par exemple). Dans ce cas, outre les erreurs et les avertissements que vous devez vérifier habituellement, vérifiez également les erreurs répertoriées dans le tableau 8. Les erreurs dans le tableau 8 indiquent que le changement d'utilisateur n'a pas abouti.

Tableau 8. Erreurs indiquant l'échec du changement d'utilisateur

SQLCODE	Signification
SQL1046N	Le contexte sécurisé qui a permis cette connexion sécurisée n'est pas configuré pour permettre le changement d'ID utilisateur que vous tentez d'effectuer. Vous ne pourrez pas utiliser cet ID autorisation tant que le contexte sécurisé n'est pas modifié.
SQL30082N	Le mot de passe entré n'est pas correct pour l'ID autorisation que vous souhaitez utiliser.
SQL0969N avec une erreur native -20361	Il existe une contrainte de niveau base de données qui vous empêche de changer d'utilisateur.

Si le changement d'utilisateur n'aboutit pas, la connexion ne sera pas établie. Vous pouvez changer d'utilisateur sur une connexion sécurisée dont l'état est non connecté mais vous ne pouvez pas accéder au serveur de bases de données. Une connexion à l'état non connecté reste en l'état jusqu'à ce que vous changiez d'utilisateur.

Remarque :

- Important :** Le fait de changer d'utilisateur sans indiquer de mot de passe supprime l'étape d'authentification du serveur de base de données. Votre application ne doit pas autoriser un changement d'ID autorisation sans mot de passe sauf si votre application a déjà validé et authentifié l'ID autorisation. Sinon, une brèche de sécurité est créée.
- Le fait d'indiquer une valeur NULL pour l'attribut SQL_ATTR_TRUSTED_CONTEXT_USERID équivaut à définir l'ID autorisation système du contexte sécurisé (l'ID utilisateur employé lors de la création de la connexion sécurisée explicite).
- Lorsque vous définissez la valeur de l'attribut de connexion SQL_ATTR_TRUSTED_CONTEXT_USERID sur une connexion sécurisée explicite, la connexion est immédiatement redéfinie. Cette redéfinition équivaut à la création d'une nouvelle connexion avec les attributs de connexion d'origine de cette connexion. Cette redéfinition a lieu même lorsque la valeur affectée à l'attribut de connexion est l'ID autorisation système ou est NULL ou est identique à celle de l'attribut.
- Si l'attribut SQL_ATTR_TRUSTED_CONTEXT_PASSWORD est défini, le mot de passe sera authentifié lors du traitement de changement d'utilisateur même si

le contexte sécurisé qui a autorisé la connexion sécurisée ne requiert d'authentification lors d'un changement d'utilisateur pour cet ID autorisation. Une surcharge système inutile survient alors. Cette règle ne s'applique pas à l'ID autorisation système du contexte sécurisé. Si l'ID autorisation système du contexte sécurisé ne requiert pas d'authentification lorsque vous souhaitez l'utiliser, alors il n'est pas authentifié même si un mot de passe est fourni.

Remarques sur l'authentification DB2 Connect

En tant qu'administrateur DB2 Connect, vous pouvez déterminer, en collaboration avec l'administrateur de votre base de données System z ou IBM Power Systems, à quel niveau seront validés les noms d'utilisateur et les mots de passe :

- Au niveau du client
- Au niveau du serveur System z ou du serveur IBM Power Systems
- Via la connexion unique ou la validation au moyen d'un système tiers (Kerberos).

Remarque : Si le client distant n'a pas spécifié de type d'authentification, le type par défaut SERVER_ENCRYPT sera utilisé. Si ce type n'est pas accepté par le serveur, le client devra effectuer une nouvelle tentative en utilisant une valeur adéquate renvoyée par le serveur. Pour optimiser les performances, spécifiez toujours le type d'authentification au niveau du client afin d'éviter ces flux de réseau supplémentaires.

A partir de DB2 Connect version 8.2.2 (équivalent à la version 8.1 FixPak 9), la passerelle n'est plus un participant passif lors de la négociation d'authentification. La passerelle a maintenant un rôle actif. Le type d'authentification indiqué dans l'entrée de répertoire de base de données sur la passerelle remplace le type d'authentification catalogué sur le client. Le client, la passerelle et le serveur doivent tous indiquer des types compatibles. Si le type d'authentification catalogué sur la passerelle n'a pas été défini dans l'entrée de répertoire de base de données, l'authentification SERVER sera le type par défaut demandé du serveur. Toutefois, la négociation aura toujours lieu entre le client et le serveur si ce dernier ne prend pas en charge l'authentification SERVER. Ce comportement est différent de celui du client qui, par défaut utilise SERVER_ENCRYPT si aucun type d'authentification n'a été indiqué.

Le type d'authentification catalogué sur la passerelle n'est pas utilisé si l'option DB2NODE ou SQL_CONNECT_NODE de l'API SET CLIENT a été définie sur le client. Dans ce cas, la négociation a lieu uniquement entre le client et le serveur.

Les types d'authentification suivants sont admis avec DB2 Connect :

CLIENT

Le nom d'utilisateur et le mot de passe sont validés par le client.

DATA_ENCRYPT

Offre la possibilité de chiffrer des données utilisateur lors de communications client/serveur. Ce type d'authentification n'est pas pris en charge sur le serveur de base de données IBM Power Systems.

KERBEROS

Permet au client de se connecter au serveur à l'aide de l'authentification Kerberos au lieu de recourir à la combinaison traditionnelle ID et mot de passe. Ce type d'authentification requiert que le client et le serveur soient tout deux activés pour l'utilisation de Kerberos.

SERVER

Le nom d'utilisateur et le mot de passe sont validés par la base de données du serveur System z ou IBM Power Systems.

SERVER_ENCRYPT

Comme pour l'authentification SERVER, le nom d'utilisateur et le mot de passe sont validés par le serveur de base de données System z ou IBM Power Systems, mais les ID utilisateur et les mots de passe transférés sont chiffrés par le client.

SERVER_ENCRYPT_AES

Les ID utilisateur et les mots de passe transférés sont chiffrés à l'aide d'un algorithme de chiffrement AES (Advanced Encryption Standard) sur le client et validés sur le serveur de base de données System z.

L'élément qui rend l'authentification Kerberos unique est que le client ne transmet pas directement d'ID utilisateur ou de mot de passe au serveur. A la place, Kerberos agit en tant que mécanisme d'authentification tiers. L'utilisateur entre une seule fois son ID et son mot de passe dans le terminal client et Kerberos valide la connexion. Ensuite, Kerberos transmet de manière automatique et sécurisée l'autorisation de l'utilisateur aux services locaux et réseau souhaités. Ainsi, l'utilisateur ne doit pas réintroduire son ID et son mot de passe sur un serveur DB2 distant. La fonction de connexion unique fournie par l'authentification Kerberos requiert que DB2 Connect et le serveur de base de données auquel il se connecte prennent tout deux en charge Kerberos.

Remarque : Le type d'authentification GSSPLUGIN n'est pas pris en charge.

Support Kerberos

La couche d'authentification Kerberos qui gère le système d'établissement de tickets est intégré dans le mécanisme Windows 2000 Active Directory. Les côtés client et serveur d'une application communiquent respectivement avec les modules client et serveur Kerberos SSP (fournisseur de support de sécurité). La SSPI (interface du fournisseur de support de sécurité) offre une interface de haut niveau au Kerberos SSP et aux autres protocoles de sécurité.

Configuration typique

Pour configurer DB2 avec l'authentification Kerberos, configurez :

- Une règle d'authentification pour DB2 (en tant que service) dans Active Directory partagée sur un réseau, et
- Une relation d'accréditation entre les centres de distribution de clés Kerberos (KDC)

Le scénario le plus simple implique la configuration d'au moins une relation d'accréditation KDC, à savoir celle entre le KDC qui contrôle le poste de travail et le système IBM Power Systems ou System z. OS/390 version 2.10 ou z/OS version 1.2 offre le traitement des tickets Kerberos via l'utilitaire RACF qui permet à l'hôte d'agir en tant que centre de distribution de clés UNIX.

DB2 Connect fournit comme d'ordinaire la fonction de routeur dans la configuration à trois niveaux. Il ne joue aucun rôle dans l'authentification lorsque la sécurité Kerberos est utilisée. Au lieu de cela, il transmet simplement le jeton de sécurité du client à DB2 for IBM i ou à DB2 for z/OS. La passerelle DB2 Connect ne doit pas être un membre du client ou du domaine Kerberos des hôtes.

Compatibilité secondaire

Configuration minimale requise DB2 pour le support Kerberos :

client IBM Data Server :

Version 8

DB2 Connect :

Version 8

DB2 for z/OS :

Version 7

Conseils et astuces relatifs à la sécurité z/OS

Cette rubrique regroupe divers conseils et astuces relatifs à la sécurité lors de la connexion de DB2 Connect à un serveur de base de données DB2 for z/OS.

Zone de sécurité étendue

Vérifiez que la zone de sécurité étendue DB2 for z/OS a été définie avec la valeur YES. Cette zone figure sous le panneau DB2 for z/OS DSNTIPR.

Codes de sécurité étendue

Jusque DB2 for z/OS version 5.1, les demandes de connexion fournissant des ID utilisateur et des mots de passe pouvaient échouer avec le message SQL30082 et le code anomalie 0, sans contenir aucune autre indication sur l'origine de l'erreur.

DB2 for z/OS version 5.1 a introduit une amélioration qui fournit une prise en charge des codes de sécurité étendue. Si la sécurité étendue est spécifiée, vous obtiendrez des diagnostics supplémentaires, tels que (PASSWORD EXPIRED) en plus du code anomalie.

Afin d'exploiter cette fonction, le paramètre d'installation DB2 for z/OS ZPARM relatif à la sécurité étendue doit être défini sur YES. Utilisez le panneau d'installation DB2 for z/OS DSN6SYSP pour définir la valeur EXTSEC=YES. Vous pouvez également utiliser le panneau DDF 1 (DSNTIPR) pour la définir. La valeur par défaut est EXTSEC=N0. Si le mot de passe a expiré, les applications Windows, Linux, UNIX, et Web utilisant DB2 Connect recevront un message d'erreur SQL30082.

Sécurité TCP/IP déjà vérifiée (option)

Si vous souhaitez fournir une prise en charge de l'option de sécurité DB2 AUTHENTICATION=CLIENT, utilisez le panneau d'installation DB2 for z/OS DSNTIP4 (panneau DDF 2) pour définir l'option Sécurité TCP/IP déjà vérifiée sur YES.

Sécurité des applications de poste de travail ODBC et Java

Les applications ODBC de poste de travail et Java utilisent le SQL dynamique. Cela peut engendrer des problèmes de sécurité sur certaines installations. DB2 for z/OS introduit une nouvelle option de définition d'accès DYNAMICRULES(BIND) qui permet l'exécution du SQL dynamique avec l'autorisation du propriétaire ou du programme de définition des accès.

DB2 et DB2 Connect fournissent un nouveau paramètre de configuration CLI/ODBC CURRENTPACKAGESET dans le fichier de configuration DB2CLI.INI. Il doit être défini sur un nom de schéma qui possède les privilèges adéquats. Une instruction SQL SET CURRENT PACKAGESET schema sera automatiquement exécutée après chaque connexion à l'application.

Utilisez le gestionnaire ODBC pour mettre à jour DB2CLI.INI.

Support de modification des mots de passe

Si le mot de passe d'un ID utilisateur a expiré, une instruction SQL CONNECT renvoie un message d'erreur, tel que SQLCODE -30082 avec le code anomalie 1. Avec DB2 Connect vous pouvez modifier le mot de passe à distance. Via l'architecture DRDA, DB2 for z/OS peut modifier le mot de passe pour vous en exécutant l'instruction CONNECT suivante :

```
CONNECT TO <basededonnées> USER <idutilisateur> USING <motdepasse>  
      NEW <nouveau_mot_de_passe> CONFIRM <nouveau_mot_de_passe>
```

La boîte de dialogue "Modification du mot de passe" de l'assistant de configuration DB2 peut également être utilisée afin de modifier le mot de passe.

Types d'authentification pris en charge avec DB2 Connect

Cette rubrique répertorie les diverses combinaisons de paramètres d'authentification et de sécurité pris en charge avec DB2 Connect.

Types d'authentification pour les connexions TCP/IP

Le protocole de communication TCP/IP ne prend pas en charge les options d'authentification au niveau du protocole réseau. Le type d'authentification détermine l'emplacement où l'authentification a lieu. Seules les combinaisons illustrées dans cette table sont prises en charge par DB2 Connect. Le paramètre d'authentification se trouve dans l'entrée de répertoire de base de données du serveur DB2 Connect.

Tableau 9. Scénarios d'authentification valides

Scénario	Paramètre d'authentification	Validation
1	CLIENT	Client
2	SERVER	Serveur de base de données grand système IBM
3	SERVER_ENCRYPT	Serveur de base de données grand système IBM
4	KERBEROS	Sécurité Kerberos
5	DATA_ENCRYPT	Hôte
6	SERVER_ENCRYPT_AES	Serveur de base de données hôte

Discussion des types d'authentification

La rubrique suivante concerne les connexions décrites ci-dessus et répertoriées dans le tableau 9, à la page 53. Chaque scénario est décrit en détail, comme suit :

- Dans le scénario 1, le nom d'utilisateur et le mot de passe sont validés uniquement au niveau de l'hôte client. Pour un client local, le nom d'utilisateur et le mot de passe sont validés uniquement au niveau du serveur DB2 Connect.

L'utilisateur est censé être authentifié à l'emplacement auquel il se connecte. L'ID utilisateur est envoyé à travers le réseau mais pas le mot de passe. Utilisez ce type de sécurité uniquement si tous les postes de travail client possèdent des fonctions de sécurité 'dignes de confiance'.

- Dans le scénario 2, le nom d'utilisateur et le mot de passe sont uniquement validés par le serveur de base de données grand système IBM. L'ID utilisateur et le mot de passe sont envoyés à travers le réseau du client distant au serveur DB2 Connect et du serveur DB2 Connect au serveur de base de données grand système IBM .
- Le scénario 3 est identique au scénario 2, à l'exception que l'ID utilisateur et le mot de passe sont chiffrés.
- Dans le scénario 4, un ticket Kerberos est obtenu du client vers Kerberos KDC. Le ticket est transféré tel quel via DB2 Connect vers le serveur, où il est validé par le serveur.
- Le scénario 5 est identique au scénario 3, à l'exception que les données utilisateur sont également chiffrées et que DATA_ENCRYPT ne prend pas en charge le serveur de base de données IBM Power Systems.
- Le scénario 6 est identique au scénario 3, à l'exception qu'un algorithme de chiffrement AES (Advanced Encryption Standard) est utilisé.

Chapitre 6. Liaison d'applications et d'utilitaires (DB2 Connect)

Les programmes d'application développés utilisant le SQL imbriqué doivent être liés aux bases de données avec lesquelles ils fonctionnent. Sur les plateformes sur lesquelles ces fonctions sont disponibles, vous pouvez définir ces liaisons à l'aide du Centre de contrôle ou de l'assistant de configuration.

Vous pouvez définir une liaison une fois par application et pour chaque base de données. Au cours du processus de liaison, les plans d'accès à la base de données sont conservés pour chaque instruction SQL exécutée. Ces plans d'accès sont fournis par les développeurs d'application et sont conservés dans les *fichiers de liens* créés au cours de la précompilation. La liaison est le processus de traitement des fichiers de liens par un serveur de base de données grand système IBM.

Comme plusieurs utilitaires fournis avec DB2 Connect sont développés à l'aide de SQL imbriqué, ils doivent être liés à un serveur de base de données grand système IBM avant de pouvoir les utiliser avec ce système. Si vous n'utilisez pas les utilitaires et interfaces de DB2 Connect, vous n'avez pas besoin de les lier à chacun de vos serveurs de base de données grand système IBM. Les listes de fichiers de liens requis par ces utilitaires se trouvent dans les fichiers suivants :

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

La liaison de ces listes de fichiers à une base de données entraîne la liaison de tous les utilitaires à cette base de données.

Si un serveur DB2 Connect est installé, les utilitaires DB2 Connect doivent être liés à chaque serveur de base de données grand système IBM avant de pouvoir les utiliser avec ce système. En supposant que les clients possèdent le même niveau de groupe de correctifs, vous devez lier les utilitaires une seule fois, indépendamment du nombre de plateformes client impliquées.

Par exemple, si 10 clients Windows et 10 clients AIX se connectent à DB2 for z/OS via DB2 Connect Enterprise Edition sur un serveur Windows, effectuez l'une des opérations suivantes :

- Liez ddcsmvs.lst à partir de l'un des clients Windows.
- Liez ddcsmvs.lst à partir de l'un des clients AIX.
- Liez ddcsmvs.lst à partir du serveur DB2 Connect.

Cet exemple suppose que :

- Tous les clients possèdent le même niveau de service. Si tel n'est pas le cas, vous devrez procéder à la liaison à partir de chaque client d'un niveau de service défini.
- Le serveur possède le même niveau de service que les clients. Si tel n'est pas le cas, vous devrez également procéder à la liaison à partir du serveur.

En outre, pour les utilitaires DB2 Connect, toute autre application utilisant le SQL imbriqué doit également être liée aux bases de données avec lesquelles vous souhaitez qu'elle fonctionne. Une application non liée engendre un message d'erreur SQL0805N lorsque vous l'exécutez. Vous souhaitez peut-être créer un fichier liste de liens supplémentaire pour vos applications qui doivent être liées.

Pour chaque serveur de base de données grand système IBM avec lequel vous établissez une liaison, procédez comme suit :

1. Vérifiez que vous possédez les droits d'accès suffisants pour le système de gestion de votre serveur de base de données grand système IBM :

System z

Les autorisations requises sont :

- SYSADM ou
- SYSCTRL ou
- BINDADD *et* CREATE IN COLLECTION NULLID

Remarque : Les privilèges BINDADD et CREATE IN COLLECTION NULLID offrent des droits d'accès suffisants **uniquement** lorsque les modules n'existent pas encore. Par exemple, si vous les créez pour la première fois.

Si les modules existent déjà et que vous les liez à nouveau, les droits d'accès requis pour effectuer le(s) tâche(s) dépendent de la personne qui a créé le lien à l'origine.

A) Si vous avez créé le premier lien et que vous en créez un nouveau, vous devez posséder l'un des droits d'accès susmentionnés pour réaliser le lien.

B) Si le premier lien a été créé par un autre utilisateur et que vous créez le second lien, vous devez posséder les droits d'accès SYSADM ou SYSCTRL pour réaliser le lien. Si vous possédez les droits BINDADD et CREATE IN COLLECTION NULLID, vous ne serez pas autorisé à créer le lien. Vous pourrez toujours créer un module si vous ne possédez pas les droits SYSADM ou SYSCTRL. Dans ce cas de figure, vous aurez besoin du privilège BIND pour chaque module existant que vous souhaitez remplacer.

VSE ou VM

L'autorisation requise est les droits d'accès DBA. Si vous souhaitez utiliser l'option GRANT de la commande Bind (pour éviter d'octroyer des accès séparés pour chaque module DB2 Connect), l'ID utilisateur NULLID doit posséder le droit d'octroyer des droits aux autres utilisateurs dans les tables suivantes :

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

Sur le système VSE ou VM, vous pouvez exécuter :
grant select on *table* to nullid with grant option

IBM Power Systems

Droits d'accès équivalents ou supérieurs aux droits *CHANGE dans la collection NULLID.

2. Exécutez des commandes similaires à la commande suivante :

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

Où *ALIAS_BD*, *ID_UTILISATEUR* et *MOT_DE_PASSE* s'appliquent au serveur de base de données grand système IBM, *ddcsmvs.lst* est le fichier de liste de liens pour z/OS et *chemin* représente l'emplacement du fichier de liste de liens.

Par exemple, *drive:\sql11b\bnd* s'applique à tous les systèmes d'exploitation Windows et *INSTHOME/sql11b/bnd/* à tous les systèmes d'exploitation Linux et UNIX, alors que *drive* représente l'unité logique sur laquelle DB2 Connect est installé et *INSTHOME* le répertoire de base de l'instance DB2 Connect.

Vous pouvez utiliser l'option *grant* de la commande *bind* pour octroyer le privilège EXECUTE à PUBLIC ou à un nom d'utilisateur ou ID de groupe spécifique. Si vous n'utilisez pas l'option *grant* de la commande *bind*, vous devez octroyer le privilège GRANT EXECUTE (RUN) individuellement.

Pour connaître les noms des modules des fichiers de liens, saisissez la commande suivante :

```
ddcspkgn @bindfile.lst
```

Par exemple :

```
ddcspkgn @ddcsmvs.lst
```

peut renvoyer le résultat suivant :

Fichier de lien	Nom du module
f:\sql11b\bnd\db2ajgrt.bnd	SQLAB6D3

Pour déterminer ces valeurs pour DB2 Connect exécutez l'utilitaire *ddcspkgn*, par exemple :

```
ddcspkgn @ddcsmvs.lst
```

Cet utilitaire peut éventuellement être utilisé pour déterminer le nom du module des fichiers de liens individuels, par exemple :

```
ddcspkgn bindfile.bnd
```

Remarque :

- a. L'utilisation de l'option de définition d'accès *sqlerror continue* est obligatoire. Cette option est cependant spécifiée automatiquement lorsque vous liez des applications à l'aide des outils DB2 ou de l'interpréteur de commandes. La spécification de cette option entraîne la transformation des erreurs en avertissements. Aussi, la liaison d'un fichier contenant des erreurs peut toujours entraîner la création d'un module. A l'inverse, elle permet l'utilisation d'un même fichier de liens dans plusieurs serveurs, même lorsque l'implémentation d'un serveur particulier entraîne le signalement d'une syntaxe SQL ou autre comme invalide. Pour cette raison, la liaison d'un des fichiers de liste *ddcsxxx.lst* à un serveur de base de données grand système IBM spécifique déclenchera probablement certains avertissements.

- b. Si vous vous connectez à une base de données DB2 via DB2 Connect, utilisez la liste de liens `db2ubind.lst`, sans spécifier `sqlerror continue`, lequel n'est valide que pour une connexion à un serveur de base de données grand système IBM. Aussi, pour vous connecter à une base de données DB2, nous vous recommandons d'utiliser les clients DB2 fournis avec DB2 et non ceux fournis avec DB2 Connect.
3. Utilisez des instructions similaires pour lier toute application ou liste d'applications.
4. Si vous possédez des clients distants issus d'une version précédente de DB2, vous devrez peut-être lier les utilitaires de ces clients à DB2 Connect.

Chapitre 7. Mises à jour multisite

La mise à jour multisite, également connue sous le nom d'unité d'oeuvre répartie (DUOW) et de validation en deux phases, est une fonction que permet aux applications de mettre à jour des données sur divers serveurs de base de données distants avec une intégrité garantie. Par exemple, une transaction bancaire impliquant le transfert d'argent d'un compte à un autre dans un serveur de base de données différent.

Pour de telles transactions, il est essentiel que ces mises à jour réalisant les opérations de débit sur un compte ne soient pas validées sans que les mises à jour requises pour traiter les crédits sur l'autre compte soient validées également. Les considérations relatives à la mise à jour multisite s'appliquent lorsque les données représentant ces comptes sont gérées par deux serveurs de base de données différents.

Les produits DB2 offrent une prise en charge globale des mises à jour multisites. Cette prise en charge est disponible pour les applications développées à l'aide du langage SQL régulier ainsi que pour les applications utilisant les moniteurs de traitement de transactions (moniteurs TP) mettant en oeuvre les spécifications de l'interface X/Open XA. Parmi les exemples de moniteurs TP : IBM TxSeries (CICS et Encina), IBM Message and Queuing Series, IBM Component Broker Series, IBM San Francisco Project ainsi que Microsoft Transaction Server (MTS), BEA Tuxedo et d'autres produits. Diverses conditions sont requises pour la configuration selon que la mise à jour multisite du SQL natif ou du moniteur TP est utilisée ou non.

Les programmes de mise à jour multisite du langage SQL natif et du moniteur TP doivent tout deux être précompilés à l'aide des options `CONNECT 2 SYNCPOINT TWOPHASE`. Les deux programmes peuvent utiliser l'instruction SQL Connect pour indiquer la base de données qu'ils souhaitent utiliser pour les instructions SQL ultérieures. Si aucun moniteur TP n'indique à DB2 qu'il va coordonner la transaction (indiqué à DB2 lorsqu'il reçoit les appels `xa_open` du moniteur TP afin d'établir une connexion à la base de données), le logiciel DB2 sera utilisé pour coordonner la transaction.

Lorsque vous utilisez la mise à jour multisite de moniteur TP, l'application doit demander la validation ou le déploiement à l'aide de l'API du moniteur TP, par exemple `CICS SYNCPOINT`, `Encina Abort()`, `MTS SetAbort()`. Lorsque vous utilisez la mise à jour multisite du langage SQL natif, les commandes SQL `COMMIT` et `ROLLBACK` habituelles doivent être utilisées.

La mise à jour multisite du moniteur TP peut coordonner une transaction qui accède à la fois à des gestionnaires de ressources DB2 et non DB2, tels qu'Oracle, Informix ou SQLServer. La mise à jour multisite du langage SQL natif est utilisée uniquement avec les serveurs DB2.

Pour qu'une transaction de mise à jour multisite fonctionne, chaque base de données participant à une transaction répartie doit être capable de prendre en charge une unité d'oeuvre répartie (DUOW). A l'heure actuelle, les serveurs DB2 prennent en charge les DUOW qui leur permettent de prendre part à des transactions réparties :

- DB2 pour Linux, UNIX et Windows version 8 ou version ultérieure

- DB2 for z/OS version 7 ou ultérieures
- DB2 for IBM i

Une transaction répartie peut mettre à jour n'importe quelle combinaison de serveurs de base de données pris en charge. Votre application peut, par exemple, mettre à jour plusieurs tables dans une base de données DB2 sous Windows, une base de données DB2 for z/OS et une base de données DB2 for i, au cours d'une seule et même transaction.

Activation des mises à jour multisite à l'aide du Centre de contrôle

Vous pouvez utiliser le Centre de contrôle pour fournir des mises à jour multisites.

Pour activer les mises à jour multisites :

1. Démarrez le Centre de contrôle.
2. Cliquez sur le signe [+] pour développer l'arborescence.
3. Avec le bouton droit de la souris, sélectionnez l'instance que vous souhaitez configurer. Un menu contextuel s'ouvre.
4. Sélectionnez l'élément de menu **Mise à jour multisite** —> **Configurer**. L'assistant de configuration de mise à jour multisite s'ouvre.
5. Sélectionnez **Utiliser le moniteur TP indiqué ici** et spécifiez un moniteur de traitement de transactions (TP). Ce champ affichera les valeurs par défaut du moniteur de traitement de transactions activé. Si vous ne souhaitez pas utiliser le moniteur de traitement de transactions, sélectionnez **Ne pas utiliser de moniteur TP**. Cliquez sur **Suivant**.
6. Si vous utilisez un moniteur TP, spécifiez les paramètres du gestionnaire de points de synchronisation. Si vous n'utilisez pas de moniteur TP, spécifiez la base de données du gestionnaire de transactions.
7. Cliquez sur **Fin**.

Test de mise à jour multisite à l'aide du Centre de contrôle

Vous pouvez tester votre configuration de mise à jour multisite à l'aide du Centre de contrôle.

Pour tester la mise à jour multisite :

1. A l'aide du bouton droit de la souris, sélectionnez l'instance et choisissez l'option de menu **Mise à jour multisite** —> **Test** dans le menu contextuel. La fenêtre Test de mise à jour multisite s'ouvre.
2. Sélectionnez les bases de données que vous souhaitez tester dans les bases de données disponibles dans la zone de liste **Objets disponibles**. Vous pouvez utiliser les flèches de direction (> et >>) situées au centre pour déplacer les sélections dans et vers la zone de liste **Objets sélectionnés**. Vous pouvez également modifier l'ID utilisateur et le mot de passe en les éditant directement dans la zone de liste **Objets sélectionnés**.
3. Lorsque votre sélection est terminée, cliquez sur **OK**. La fenêtre Résultats du test de mise à jour multisite s'ouvre.

4. La fenêtre Résultats du test de mise à jour multisite indique les bases de données sélectionnées qui ont réussi ou non le test de mise à jour. La fenêtre indique les codes SQL et les messages d'erreur des tests non réussis. Cliquez sur **Fermeture** pour fermer la fenêtre.
5. Cliquez sur **Fermeture** pour fermer la fenêtre Test de mise à jour multisite.

Mise à jour multisite et gestionnaire de points de synchronisation

Les serveurs de base de données grand système IBM requièrent DB2 Connect pour prendre part à une transaction répartie provenant d'applications Linux, Windows, UNIX et Web. De plus, plusieurs scénarios de mise à jour multisite impliquant des serveurs de base de données grand système IBM imposent la configuration du gestionnaire de points de synchronisation (SPM). Lorsqu'une instance DB2 est créée, le gestionnaire de points de synchronisation (SPM) DB2 est configuré automatiquement avec les valeurs par défaut.

Le besoin en SPM est dicté par le choix du protocole (TCP/IP) et l'utilisation d'un moniteur TP. Le tableau suivant fournit un récapitulatif des scénarios qui requièrent l'utilisation du SPM. Le tableau indique également si DB2 Connect est requis pour accéder au serveur grand système IBM depuis des machines Intel® ou UNIX. Pour les mises à jour multisites, le composant SPM de DB2 Connect est requis si vous utilisez le moniteur TP.

Tableau 10. Scénarios de mise à jour multisite requérant le SPM – TCP/IP

Moniteur TP utilisé ?	Gestionnaire de points de synchronisation nécessaire ?	Produit requis (choisissez-en un)	Base de données grand système IBM prise en charge
Oui	Oui	Serveur DB2 Connect DB2 Enterprise Server Edition avec licence DB2 Connect appliquée	DB2 for z/OS V7 DB2 for z/OS V8 ou ultérieure
Non	Non	DB2 Connect Personal Edition Serveur DB2 Connect DB2 Enterprise Server Edition avec licence DB2 Connect appliquée	DB2 for z/OS V7 DB2 for z/OS V8 ou ultérieure

Remarque : Une transaction répartie peut mettre à jour n'importe quelle combinaison de serveurs de base de données pris en charge. Votre application peut, par exemple, mettre à jour plusieurs tables dans une base de données DB2 sous Windows, une base de données DB2 for z/OS et une base de données DB2 for IBM i, au cours d'une seule et même transaction.

Configuration de DB2 Connect avec un gestionnaire de transactions compatible XA

Cette rubrique décrit les étapes de configuration requises pour utiliser des serveurs de base de données IBM Power Systems et System z dans votre moniteur TP.

Vous devez disposer d'un moniteur TP opérationnel et DB2 Connect doit être installé ; vous devez également avoir configuré et testé une connexion au serveur de base de données grand système IBM.

Pour configurer DB2 Connect afin d'utiliser des serveurs de base de données IBM Power Systems et System z dans votre moniteur TP, procédez comme suit :

1. Configurez le moniteur TP de façon à ce qu'il puisse accéder au commutateur XA DB2. Le commutateur XA DB2 fournit au moniteur TP les adresses des API XA de DB2 Connect. Chaque moniteur TP possède son propre procédé pour effectuer cette opération.
2. Configurez le moniteur TP avec une chaîne DB2 XA_OPEN. Chaque moniteur TP possède son propre procédé pour effectuer cette opération. Pour obtenir des informations sur la configuration d'une chaîne DB2 XA OPEN en vue de son utilisation par le moniteur TP, consultez la documentation du moniteur TP.
3. Si nécessaire, modifiez les paramètres de configuration par défaut du gestionnaire de points de synchronisation DB2 Connect. Les serveurs de base de données hôte IBM et System i (version 5 édition 3 et antérieures) ne prennent pas encore en charge l'interface XA. System i Version 5 Release 4 et les versions suivantes prennent entièrement en charge XA.

Le SPM est un composant DB2 Connect qui mappe le protocole de validation en deux phases XA à celui utilisé par les serveurs de base de données grand système IBM. Par défaut, l'instance DB2 possède des valeurs prédéfinies pour les paramètres de configuration SPM. Le paramètre le plus important est le paramètre de configuration du gestionnaire de base de données SPM_NAME. Il est défini par défaut sur une variante des sept premiers caractères du nom d'hôte TCP/IP.

Si vous utilisez le protocole TCP/IP pour vous connecter à DB2 for z/OS, vous ne devriez pas avoir à modifier un paramètre par défaut quelconque. Dans ce cas de figure, aucune configuration SPM n'est requise car il s'agit déjà d'un paramètre optionnel.

Prise en charge par DB2 Connect des transactions à couplage lâche

Le support de DB2 Connect pour les transactions à couplage est destiné aux utilisateurs qui implémentent des applications XA réparties accédant à DB2 for IBM i version 5 édition 4 (ou ultérieure) et à DB2 for z/OS version 7 (ou ultérieure). Cette prise en charge permet à diverses branches de la même transaction globale de partager l'espace de verrouillage sur DB2 for z/OS.

La prise en charge des transactions à couplage lâche est conçue pour les applications .NET et COM+ uniquement.

Cette fonction réduit la fenêtre dans laquelle un branchement de la transaction répartie rencontre un délai d'attente de verrouillage ou un blocage en raison de la présence d'un autre branchement au sein de la même transaction globale.

Chapitre 8. Déplacement de données avec DB2 Connect

Si vous travaillez dans un environnement complexe dans lequel vous devez déplacer des données entre un système de base de données hôte et un poste de travail, vous pouvez utiliser DB2 Connect, la passerelle de transfert de données entre l'hôte et le poste de travail (voir figure 8).

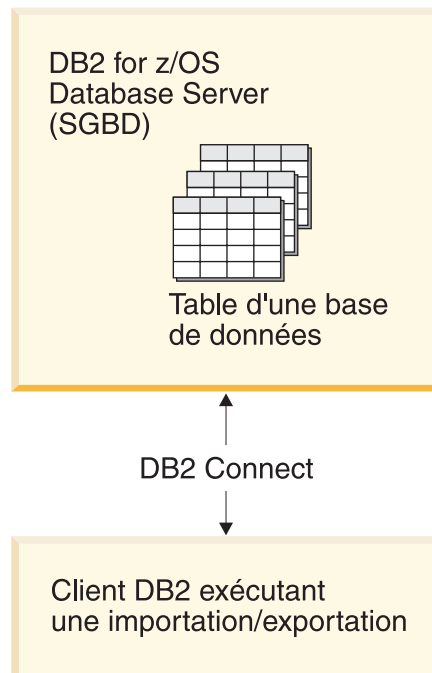


Figure 8. Importation/exportation via DB2 Connect

Les utilitaires DB2 d'exportation et d'importation vous permettent de transférer des données d'une base de données d'un serveur grand système IBM vers un fichier sur le poste de travail DB2 Connect et inversement. Vous pouvez ensuite utiliser les données avec n'importe quel système de gestion de base de données relationnelle ou n'importe quelle application compatible avec ce format d'exportation ou d'importation. Par exemple, vous pouvez exporter des données à partir d'une base de données d'un serveur grand système IBM vers un fichier PC/IXF, puis l'importer dans une base de données DB2 Database for Linux, UNIX, and Windows.

Vous pouvez exécuter les opérations d'importation et d'exportation à partir d'un client de base de données ou du poste de travail DB2 Connect.

Remarque :

1. Les données à exporter ou importer doivent être conformes aux restrictions en termes de taille et de type de données s'appliquant aux deux bases de données.

2. Pour améliorer les performances des l'importation, vous pouvez utiliser des requêtes composées. Indiquez le modificateur de type de fichier compound dans l'utilitaire d'importation pour regrouper un nombre précis d'instructions de requêtes dans un bloc. Ceci réduit la charge du réseau et améliore les temps de réponse.

Avec DB2 Connect, les opérations d'importation et d'exportation doivent respecter les conditions suivantes :

- Le type de fichier doit être PC/IXF.
- Vous devez créer une table cible avec des attributs compatibles avec les données sur le serveur cible avant de procéder à l'importation. Vous pouvez employer l'utilitaire db2look pour obtenir les attributs de la table source. L'importation via DB2 Connect ne permet pas de créer une table, car INSERT est la seule option prise en charge.

Si l'un de ces conditions n'est pas satisfaite, l'opération échoue et un message d'erreur est renvoyé.

Remarque : Les définitions d'index ne sont pas stockées lors de l'exportation ou utilisées lors de l'importation.

Si vous exportez ou importez des données mixtes (des colonnes contenant à la fois des données SBCS et DBCS c'est-à-dire codées sur un seul octet et sur deux octets) prenez en compte les considérations suivantes :

- Sur les systèmes stockant les données en EBCDIC (MVS, System z, IBM Power Systems, VM et VSE), des caractères de code normal et de code DBCS marquent le début et la fin des données DBCS. Lorsque vous définissez les longueurs de colonnes de vos tables de table de base de données, pensez à prévoir assez d'espace pour ces caractères.
- Les colonnes de caractères de longueur variable sont recommandées, sauf si les données de la colonne ont un canevas régulier.

Déplacement de données d'un poste de travail vers un serveur hôte

Pour déplacer des données vers une base de données de serveur hôte ou System i :

1. Exportez les données à partir d'une table DB2 vers un fichier PC/IXF.
2. A l'aide de l'option INSERT, importez le fichier PC/IXF dans une table compatible dans la base de données du serveur hôte.

Pour transférer des données d'une base de données du serveur hôte vers un poste de travail :

1. Exportez les données de la table de la base de données du serveur hôte vers un fichier PC/IXF.
2. Importez le fichier PC/IXF dans une table DB2.

Exemple

L'exemple suivant monte comment déplacer des données d'un poste de travail vers une base de données d'un système hôte ou d'un serveur System i.

Exportez les données dans un format IXF externe à l'aide de la commande suivante :

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

Lancez la commande suivante pour établir une connexion DRDA à la base de données cible DB2 :

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

Si elle n'existe pas déjà, créez la table cible sur l'instance de base de données cible DB2 :

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

Pour importer les données, lancez la commande suivante :

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

Chaque ligne de données est lu à partir du fichier au format IXF, puis une instruction SQL INSERT est émise pour insérer la ligne dans la table mydb.staff. Des lignes individuelles continuent d'être insérées jusqu'à ce que toutes les données aient été déplacées dans la table cible.

Vous trouverez des informations détaillées dans "Moving Data Across the DB2 Family," une publication IBM Redbooks. Vous pourrez trouver ce Redbooks publication à l'adresse URL suivante : <http://www.redbooks.ibm.com/redbooks/SG246905>.

Chapitre 9. Mappage SQLCODE

Diverses bases de données relationnelles IBM ne produisent pas toujours les mêmes codes SQLCODE pour les mêmes erreurs. Même si le SQLCODE est identique, il peut être accompagné de jetons spécifiés de manière différente. La liste des jetons est transférée dans le champ SQLERRMC de SQLCA. Par défaut, DB2 Connect mappe les SQLCODE et les jetons de chaque serveur de base données grand système IBM système vers les SQLCODE DB2 appropriés.

Si vous souhaitez désactiver le mappage de codes SQLCODE, spécifiez **NOMAP** dans la chaîne de paramètres du répertoire DCS.

Si vous portez une application directement d'un serveur de base de données grand système IBM, tel que DB2 for z/OS, il peut être souhaitable de désactiver le mappage SQLCODE. Ainsi, vous pouvez utiliser l'application sans modifier les SQLCODE qu'elle référence.

Désactivation du mappage SQLCODE

Si vous souhaitez désactiver le mappage de codes SQLCODE, spécifiez **NOMAP** dans la chaîne de paramètres du répertoire DCS.

Si vous portez directement une application depuis un serveur de base de données grand système IBM, tel que DB2 for z/OS, il peut être souhaitable de désactiver le mappage SQLCODE. Ainsi, vous pouvez utiliser l'application sans modifier les SQLCODE qu'elle référence.

Personnalisation du mappage SQLCODE

Par défaut, DB2 Connect mappe les SQLCODE et les jetons de chaque serveur de base données grand système IBM vers les SQLCODE DB2 appropriés. Les fichiers suivants sont des copies du mappage SQLCODE par défaut :

- dcs1dsn.map mappe les SQLCODE DB2 for z/OS.
- dcs1ari.map mappe les SQLCODE DB2 Server for VM and VSE.
- dcs1qsq.map mappe les SQLCODE DB2 for IBM i.

Aucun mappage n'est nécessaire pour DB2 sur les systèmes d'exploitation Linux ou UNIX.

1. Si vous désirez remplacer le mappage SQLCODE par défaut ou si vous utilisez un serveur de base de données grand système IBM dépourvu de mappage SQLCODE (un serveur de base de données non IBM), vous pouvez copier l'un de ces fichiers et l'utiliser comme base pour votre nouveau fichier de mappage SQLCODE. En procédant à la copie du fichier au lieu de l'éditer directement, vous êtes ainsi assuré de toujours pouvoir vous référer au mappage SQLCODE original, en cas de besoin.
2. Spécifiez le nom de fichier de votre nouveau fichier de mappage SQLCODE dans la chaîne de paramètres du répertoire DCS.
3. Chaque fichier de mappage est un fichier ASCII créé et édité à l'aide d'un éditeur ASCII. Lors de l'installation initiale, le fichier est stocké dans le répertoire map du chemin d'installation.

Le fichier peut contenir les types de ligne spéciaux suivants :

- &&** Le début logique d'un fichier. Toutes les lignes situées avant cette première occurrence && sont considérées comme des commentaires à format libre et sont ignorées. Si le fichier est vide après les &&, aucun mappage SQLCODE ne sera effectué. Vous pouvez également désactiver le mappage SQLCODE à l'aide du paramètre NOMAP, comme indiqué précédemment.
- *** Quand il s'agit du premier caractère d'une ligne, il indique qu'il s'agit d'un commentaire.
- W** Lorsqu'il s'agit du seul caractère indiqué dans une ligne, il indique que les options d'avertissement doivent à nouveau être mappées. Par défaut, les options d'avertissement d'origine sont transmises. Le W doit être écrit en majuscule.

Toutes les lignes situées après les && doivent être vides ou des instructions de mappage qui ont la forme suivante :

```
code_entrée  
[, code_sortie [, liste_jeton]]
```

Le *code_entrée* représente l'un des codes suivants :

sqlcode SQLCODE du serveur de base de données grand système IBM.

- U** Tous les SQLCODE négatifs non définis (ceux qui ne sont pas répertoriés dans ce fichier) sont mappés au *code_sortie* spécifié. Si aucun *code_sortie* n'est spécifié sur cette ligne, le SQLCODE original est utilisé. Ce caractère doit être indiqué en majuscule.
- P** Tous les SQLCODE positifs non définis (ceux qui ne sont pas répertoriés dans ce fichier) sont mappés au *code_sortie* spécifié. Si aucun *code_sortie* n'est spécifié sur cette ligne, le SQLCODE original est utilisé. Ce caractère doit être indiqué en majuscule.
- ccnn** Code de classe SQLSTATE du serveur de base de données grand système IBM. *nn* est l'une des valeurs suivantes :

- 00 Exécution terminée normalement
- 01 Avertissement
- 02 Pas de données
- 21 Violation de cardinalité
- 22 Condition d'exception de données
- 23 Violation de contrainte
- 24 Etat de curseur incorrect
- 26 Identificateur d'instruction SQL incorrect
- 40 Annulation de transaction (ROLLBACK)
- 42 Violation d'accès
- 51 Etat d'application incorrect
- 55 Objet non disponible dans l'état prérequis
- 56 Erreurs diverses SQL ou du produit
- 57 Ressource non disponible ou intervention d'un opérateur
- 58 Erreur système

Le *code_sortie* spécifié est utilisé pour tous les SQLCODE possédant ce code de classe qui ne sont pas spécifiés explicitement dans le fichier de mappage. Si aucun *code_sortie* n'est spécifié sur cette ligne, le SQLCODE original est mappé à lui-même et aucun jeton ne sera copié.

Les caractères **cc** doivent être indiqués en minuscule.

Si le même *code_entrée* apparaît plusieurs fois dans le fichier de mappage, la première occurrence est utilisée. Le *code_sortie* représente le SQLCODE de sortie. Si aucune valeur n'est spécifiée, le SQLCODE original est utilisé.

Si vous spécifiez un code de sortie, vous pouvez également spécifier l'une des options suivantes :

- (s) Le SQLCODE en entrée et l'ID de produit (ARI, DSN ou QSQ) seront placés dans le champ de jeton de message SQLCA.

Le SQLCODE original est renvoyé en tant que jeton unique. Cette option est conçue pour gérer des SQLCODE non définis, à l'exception des codes +965 et -969. Si +965 ou -969 représente le *code_sortie*, la liste de jetons renvoyée dans le champ SQLERRMC de SQLCA comprend le SQLCODE d'origine suivi de l'identificateur de produit et de la liste de jetons d'origine.

Le caractère **s** doit être indiqué en minuscule.

(*liste_jetons*)

Une liste de jetons, séparés par des virgules. Spécifiez une seule virgule pour passer à un jeton spécifique. Par exemple, le format (*t2,,t4*) signifie que le premier et le troisième jeton ont une valeur null.

Chaque jeton a la forme d'un numéro (*n*), précédé, facultativement, de la lettre **c** et suivi, facultativement, de la lettre **c** ou **i**. Il est interprété comme suit :

- c** Le type de données du jeton situé à cet endroit est CHAR (le type par défaut). Si la lettre **c** est spécifiée avant la lettre *n*, elle se réfère à un jeton d'entrée ; si elle est spécifiée après la lettre *n*, elle indique un jeton de sortie. Le caractère **c** doit être indiqué en minuscule.
- i** Le type de données du jeton situé à cet endroit est INTEGER. Si la lettre **i** est spécifiée après la lettre *n*, elle indique un jeton de sortie. La lettre **i** ne doit pas figurer avant *n* étant donné que les serveurs de base de données grand système IBM ne prennent en charge que les jetons de type CHAR. Le caractère **i** doit être indiqué en minuscule.
- n* Numéro ou numéros indiquant les jetons de serveur de base de données grand système IBM utilisés. Ils sont classés dans l'ordre souhaité afin d'être placé dans la SQLCA de sortie. Le numéro indique le jeton du serveur de base de données grand système IBM ; l'agencement indique l'ordre dans lequel seront placés les jetons dans la SQLCA.

Par exemple, le serveur de base de données grand système IBM peut renvoyer deux jetons, 1 et 2. Si vous souhaitez que le jeton 2 apparaisse avant le jeton 1 dans la sortie SQLCA, spécifiez (2,1).

De nombreux numéros de jeton peuvent ainsi être combinés pour former un jeton de sortie CHAR en les connectant par périodes.

Les virgules sont utilisées pour séparer les jetons de sortie. Si aucun jeton n'est spécifié avant une virgule, aucun jeton de sortie ne sera inclus dans la SQLCA à cet emplacement. Tout jeton apparaissant dans la SQLCA de sortie et qui suit le dernier jeton spécifié, est mappé à un jeton de valeur nulle.

La figure 9 indique un exemple de fichier de mappage SQLCODE.

```

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

-30021 , -30021 , (c1c,c2c)

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

```

Figure 9. Un fichier de mappage SQLCODE

Les descriptions suivantes correspondent au numéro de ligne correspondant dans la figure précédente :

1. Le SQLCODE est mappé de -007 vers -007. Le premier jeton d'entrée reçu du serveur de base de données grand système IBM est utilisé comme premier jeton de sortie et son type est défini par défaut à CHAR. Aucun autre jeton n'est transféré.
2. Le SQLCODE est mappé de -010 vers -010 (aucun SQLCODE de sortie n'est spécifié). Aucun jeton n'est placé dans la SQLCA de sortie.
3. Le SQLCODE est mappé de -060 vers -171. Le premier jeton d'entrée reçu du serveur de base de données grand système IBM est ignoré. Le second jeton est utilisé en tant que premier jeton dans la SQLCA de sortie et possède le type CHAR. Aucun second jeton n'est présent dans la SQLCA de sortie.
4. Le SQLCODE est mappé de -204 vers -204. Le premier et le second jeton reçus du serveur de base de données grand système IBM sont de type CHAR. Ces deux jetons d'entrée sont combinés afin de former un seul jeton de sortie CHAR qui deviendra le premier jeton de sortie dans la SQLCA.
5. Le SQLCODE est mappé de -633 vers -206. Le premier jeton d'entrée reçu du serveur de base de données grand système IBM est de type CHAR. Il est converti en type INTEGER et est utilisé en tant que second jeton dans la SQLCA. Le premier jeton dans la SQLCA est null, comme indiqué par la virgule.
6. Le SQLCODE est mappé de -30021 vers -30021. Le premier et le second jeton reçus du serveur de base de données grand système IBM sont de type CHAR et sont utilisés en tant que premier et second jetons dans la sortie SQLCA.
7. Tous les SQLCODE des SQLCA avec des SQLSTATE dans la classe 00 seront mappés vers SQLCODE +000.

8. Tous les SQLCODE non définis sont mappés vers -969. Cette option doit uniquement être utilisée lorsque tous les codes pouvant être mappés sont répertoriés, notamment les codes identiques qui ne requièrent aucun mappage. L'option **(s)** indique que la liste de jetons à renvoyer dans le champ SQLERRMC de la SQLCA inclut le SQLCODE d'origine, suivi du produit dans lequel l'erreur s'est produite et de la liste de jetons d'origine. Si l'entrée **U** n'est pas incluse, tous les codes non répertoriés sont transmis sans aucun mappage.
9. Tous les SQLCODE positifs non définis sont mappés vers +965. Cette option doit uniquement être utilisée lorsque tous les codes pouvant être mappés sont répertoriés, notamment les codes identiques qui ne requièrent aucun mappage. L'option **(s)** indique que la liste de jetons à renvoyer dans le champ SQLERRMC de la SQLCA inclut le SQLCODE d'origine, suivi du produit dans lequel l'avertissement est apparu et de la liste de jetons d'origine. Si l'entrée **P** n'est pas incluse, tous les codes positifs non répertoriés sont transmis sans aucun mappage.

Chapitre 10. Surveillance du système de base de données et DB2 Connect

Ce chapitre aborde plusieurs méthodes de surveillance des connexions et des performances dans un environnement utilisant DB2 Connect. Le type de surveillance effectué est spécifique au système d'exploitation.

Contrôle des connexions des clients éloignés

Vous pouvez utiliser le moniteur du gestionnaire de bases de données avec un serveur DB2 Connect, tel que DB2 Connect Enterprise Edition, pour gérer les connexions client distantes. Pour gérer les clients locaux sur le serveur DB2 Connect fonctionnant sur le serveur, vous devez définir la variable suivante :

```
db2set DB2CONNECT_IN_APP_PROCESS=NO
```

Par exemple, lorsqu'une erreur se produit au niveau du serveur grand système IBM, son administrateur peut déterminer si l'incident émane du poste de travail DB2 Connect. Le moniteur du gestionnaire de base de données correspond :

- Le jeton de corrélation DRDA (CRRTKN), pour les conversations non protégées.
- à l'ID de l'unité d'oeuvre (UOWID), pour les connexions à deux phases protégées par le gestionnaire de points de synchronisation DRDA-3 (utilisé dans les connexions TCP/IP).
- à l'ID connexion DB2 Connect (l'ID application).

Ces informations illustrent la connexion DB2 Connect à l'origine de l'incident, ce qui permet à l'administrateur système de forcer une application client individuelle du système sans affecter d'autres clients à l'aide de la connexion DB2 Connect.

Liste des états des commutateurs de contrôle

Pour répertorier l'état des commutateurs de contrôle, utilisez la commande `db2 get monitor switches`.

Contrôle des performances à l'aide du moniteur de performances de Windows

Les systèmes d'exploitation Windows offrent un outil de gestion des performances de vos applications DB2. Le moniteur de performances, l'un des outils d'administration Windows, affiche une représentation graphique des performances système. Vous pouvez choisir divers systèmes, bases de données ou éléments de communication pour les contrôler et les mapper dans une représentation graphique.

Par exemple, vous pouvez tracer en temps réel les graphiques des rapports disponibles via les commandes `GET SNAPSHOT FOR ALL DCS DATABASES` ou `GET SNAPSHOT FOR ALL DCS APPLICATIONS` à l'aide du moniteur et les comparer directement à des valeurs, telles que l'utilisation de l'unité centrale. Vous pouvez comparer directement les effets de divers paramètres sur les performances de la base de données et des communications. Vous pouvez sauvegarder vos configurations spécifiques des paramètres dans des fichiers PMC que vous pouvez extraire ultérieurement.

Par exemple dans la figure suivante, vous pouvez tracer le graphique de diverses mesures DB2 par rapport à l'utilisation de l'unité centrale. Les valeurs représentées sous forme graphique ont été enregistrées dans le fichier db2chart.pmc. Vous pouvez enregistrer autant de fichiers PMC que vous le souhaitez, chacun reflétant une coupe différente des performances système.

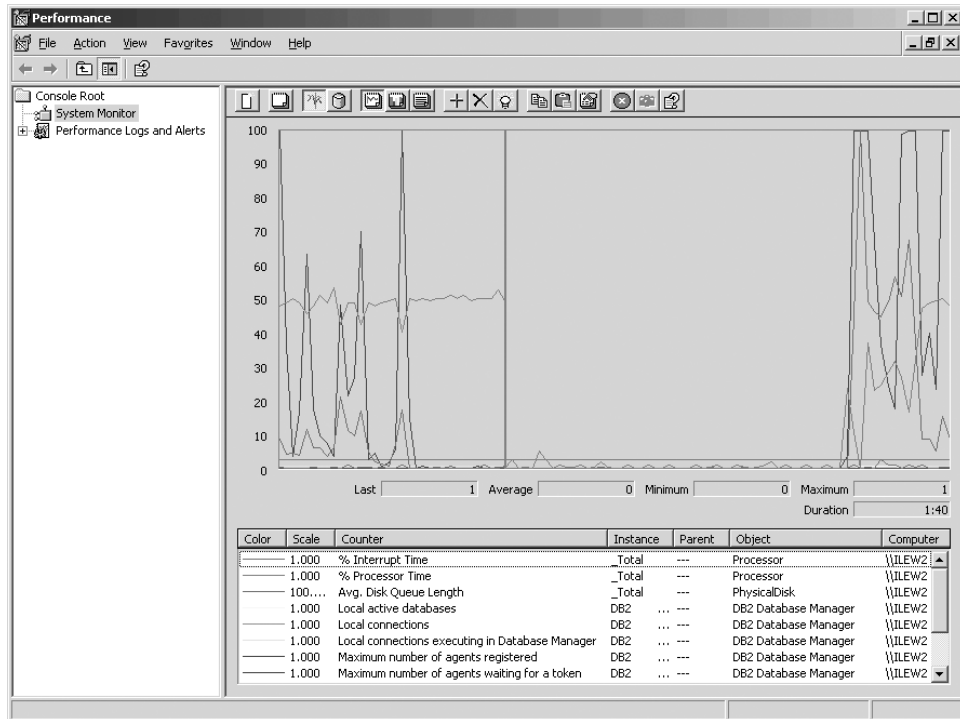


Figure 10. Moniteur de performances

Pour activer le contrôle des applications locales, vous devez désactiver la variable d'environnement DB2CONNECT_IN_APP_PROCESS.

Utilisation des commandes GET SNAPSHOT

Le moniteur DB2 gère un recueil d'informations système importantes. Vous pouvez obtenir un récapitulatif de l'état système à n'importe quel moment en exécutant la commande GET SNAPSHOT. Vous pouvez prendre des images instantanées du moniteur si vous possédez les droits d'accès SYSMAINT, SYSCTRL ou SYSADM pour l'instance gestionnaire de bases de données que vous souhaitez contrôler.

Il existe cinq commandes de prise d'image instantanée utiles au contrôle des informations DCS. Il s'agit des commandes suivantes :

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

Chaque commande de prise d'image instantanée produit un rapport détaillé sur la zone concernée.

Par exemple, la commande GET SNAPSHOT FOR DCS DATABASE ON DCSDB génère le rapport suivant :

Image instantanée de la base de données DCS

```

Nom de la base de données                = DCSDB
Nom de la base de données hôte          = GILROY
Horodatage de la première connexion à la base = 12-15-2001 10:28:24.596495
Durée d'établissement dernière connexion = 0.950561
Durée de la dernière connexion          = 0.000000
Temps de réponse hôte (sec.ms)          = 0.000000
Horodatage de la dernière réinitialisation =
Tentatives d'émission d'instructions SQL = 2
Instructions SQL COMMIT émises           = 1
Instructions SQL ROLLBACK émises         = 0
Instructions SQL ayant échouées          = 0
Nombre total de connexions passerelle    = 1
Nombre actuel de connexions passerelle   = 1
Connex. passerelle en attente réponse hôte = 0
Connex. passerelle attente demande client = 1
Erreurs de communication passerelle/hôte = 0
Horodatage dernière erreur de communication = Aucun
Cote d'alerte haute connexions passerelle = 1
Lignes sélectionnées                     = 0
Octets sortants envoyés                   = 140
Octets sortants reçus                     = 103

```

Ce rapport fournit des informations sur les connexions à la base de données, les performances, les erreurs et le débit des requêtes SQL. Les instantanés du moniteur DB2 peuvent en fait être beaucoup plus détaillés. Par exemple, si vous exécutez la commande GET SNAPSHOT FOR ALL DCS APPLICATIONS, vous recevez un rapport similaire au rapport suivant :

Image instantanée de l'application DCS

```

ID application client                    = 09150F74.B6A4.991215152824
Numéro de séquence                      = 0001
ID autorisation                          = SMITH
Nom de l'application                     = db2bp
Descripteur de l'application             = 1
Etat de l'application                    = en attente de demande
Horodatage du changement d'état          = 12-15-2001 10:29:06.707086
Noeud client                             = sys143
Niveau d'édition client                  = SQL06010
Plateforme client                        = AIX
Protocole client                         = TCP/IP
Page de codes client                     = 850
ID traitement de l'application client     = 49074
ID de connexion client                   = smith
ID application hôte                       = G9150F74.B6A5.991215152825
Numéro de séquence                       = 0000
Alias de base de données au niveau de la passerelle = MVSDB
Nom de la base de données                 = DCSDB
Nom de la base de données hôte           = GILROY
Niveau d'édition hôte                    = DSN05012
CCSID hôte                               = 500

Adresse de communication sortante         = 9.21.21.92 5021
Protocole communication sortant           = TCP/IP
Adresse communication entrante            = 9.21.15.116 46756
Horodatage de la première connexion à la base = 12-15-2001 10:28:24.596495
Temps de réponse hôte (sec.ms)           = 0.000000
Durée du traitement de la passerelle     = 0.000000
Horodatage de la dernière réinitialisation =
Lignes sélectionnées                     = 0
Tentatives d'émission d'instructions SQL = 2
Instructions SQL ayant échouées           = 0

```

```

Nombre de COMMIT = 1
Nombre de ROLLBACK = 0
Octets entrants reçus = 404
Octets sortants envoyés = 140
Octets sortants reçus = 103
Octets entrants envoyés = 287
Nombre de curseurs actifs = 0
Durée d'inactivité de l'application = 1 minute et 32 secondes

Etat d'avancement de l'unité d'oeuvre =
Horodatage de fin de l'unité d'oeuvre précédente = 12-15-2001 10:28:25.592631
Horodatage du début de l'unité d'oeuvre = 12-15-2001 10:29:06.142790
Horodatage d'achèvement de l'unité d'oeuvre =
Durée dernière unité d'oeuvre exécutée (s) = 0.034396

Opération la plus récente = Exécution immédiate
Horodatage de début de l'opération la plus récente = 12-15-2001 10:29:06.142790
Horodatage de fin de l'opération la plus récente = 12-15-2001 10:29:06.707053

Instruction = Exécution immédiate
Numéro de section = 203
Auteur de l'application = NULLID
Nom du module = SQLC2C07
Estimation du coût en timerons par compilateur SQL = 0
Estimation de la cardinalité par compilateur SQL = 0
Horodatage du début de l'instruction = 12-15-2001 10:29:06.142790
Horodatage de la fin de l'instruction = 12-15-2001 10:29:06.707053
Temps de réponse hôte (sec.ms) = 1.101612
Durée dernière instruction exécutée (s) = 0.564263
Lignes extraites = 0
Durée du traitement de la passerelle = 0.013367
Octets entrants reçus pour l'instruction = 220
Octets sortants envoyés pour l'instruction = 130
Octets sortants reçus pour l'instruction = 49
Octets entrants envoyés pour l'instruction = 27
Libellé de l'instruction SQL :
create table t12 (col1 int, col2 char)

```

Etat de l'application DCS

Le moniteur système fournit trois formes de la commande LIST DCS APPLICATIONS :

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

Dans la sortie qui s'ensuit, le format de l'ID d'application hôte et de l'ID d'application client peut varier en fonction de la version de la base de données grand système IBM et du niveau de prise en charge de TCP/IP.

Tableau 11. Format de l'ID d'application en fonction de la version de l'hôte et du niveau de support TCP/IP

Scénario	Format de l'ID d'application
Clients accédant aux serveurs de données avec un support de niveau du gestionnaire de base de données relationnelle inférieur à 7	G91A0D3A.P8BC.060306212019

Tableau 11. Format de l'ID d'application en fonction de la version de l'hôte et du niveau de support TCP/IP (suite)

Scénario	Format de l'ID d'application
Clients accédant aux serveurs de données avec un support de niveau du gestionnaire de base de données relationnelle égal ou supérieur à 8 sur le protocole TCP/IP v4	9.26.13.61.65289.060306213816
Clients accédant aux serveurs de données avec un support de niveau du gestionnaire de base de données relationnelle égal ou supérieur à 8 sur le protocole TCP/IP v6	2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741

LIST DCS APPLICATIONS

Pour visualiser les informations fournies par le moniteur au niveau d'application, exécutez la commande DB2 LIST DCS APPLICATIONS.

Elle renvoie les informations suivantes sur une connexion TCP/IP (DB2 Connect vers DB2 for z/OS):

```
Auth Id Application Name Appl. Host Application Id
                        Handle
-----
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 d'autorisation utilisé pour se connecter au serveur de base de données grand système IBM. Il identifie la personne qui exécute l'application.

Application Name

Le nom de l'application fonctionnant sur le client connue par DB2 Connect. Seules les 20 premiers octets situés après le dernier séparateur de chemin d'accès sont disponibles.

Appl. Handle

L'agent en cours d'exécution sur le poste de travail DB2 Connect. Vous pouvez utiliser cet élément pour lier les informations relatives au moniteur du gestionnaire de bases de données aux autres données de diagnostic. L'ID agent est également requis lorsque vous utilisez la commande ou l'API FORCE USERS.

ID application hôte

L'un des ID suivants :

- Le jeton de corrélation DRDA (CRRTKN), pour les conversations non protégées.

- L'ID de l'unité d'oeuvre (UOWID), pour les connexions à deux phases protégées par le gestionnaire de points de synchronisation DRDA-3 (utilisé dans les connexions TCP/IP).

Cet identifiant unique est généré lorsque l'application se connecte au serveur de base de données grand système IBM. Vous pouvez utiliser cet élément conjointement avec l'Application ID afin de mettre en corrélation les parties client et serveur des informations de l'application.

LIST DCS APPLICATIONS SHOW DETAIL

Si le format de commande DB2 LIST DCS APPLICATIONS SHOW DETAIL est spécifié, des informations supplémentaires s'affichent, notamment :

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

Identifie de manière unique l'application connectée au poste de travail DB2 Connect. Il existe différents formats pour l'ID application qui dépendent du protocole de communication établi entre le client et le poste de travail DB2 Connect.

Cette valeur vous permet de mettre en corrélation des connexions établies depuis les clients vers le poste de travail DB2 Connect et depuis le poste de travail DB2 Connect vers le serveur de base de données grand système IBM.

Client Sequence no (Seq#)

Le numéro de séquence client est le numéro de séquence de transaction. Il est utilisé pour la mise en corrélation d'une transaction répartie sur divers systèmes.

Client DB alias

L'alias de base de données fourni par l'application pour se connecter à la base de données. Cet élément peut être utilisé pour identifier la base de données actuelle à laquelle l'application accède. Le mappage entre ce nom et le nom de la base de données peut être effectué à l'aide des répertoires de base de données du noeud client et du noeud serveur gestionnaire de bases de données.

Client NNAME (Node)

Identifie le noeud sur lequel l'application client s'exécute. Les informations varient en fonction du protocole client utilisé. Pour un client connecté au moyen du protocole TCP/IP, il s'agit du nom d'hôte.

Client Product ID (Client)

Le produit et la version qui fonctionnent sur le client. Les ID du produit client seront :

- SQL07010 pour la version 7.1 de DB2 Universal Database et de DB2 Connect et de leurs clients.
- SQL08010 pour la version 8.1 de DB2 Universal Database et de DB2 Connect et de leurs clients.
- SQL08020 pour la version 8.2 de DB2 Universal Database et DB2 Connect et de leurs clients.
- SQL09120 pour les produits DB2 version 9.1, les produits DB2 Connect et leurs clients.

Code Page ID

L'identifiant de la page de codes au niveau du noeud sur lequel l'application surveillée est lancée.

Vous pouvez utiliser ces informations afin de vérifier que la conversion des données est prise en charge entre la page de codes de l'application et celle de la base de données (ou pour les bases de données de serveur grand système IBM, le CCSID du serveur de base de données grand système IBM).

Si la page de codes de l'application diffère de la page de codes avec laquelle le moniteur du gestionnaire de bases de données fonctionne, cet élément de page de codes vous aide à convertir manuellement les données transmises par l'application et affichées dans le moniteur du gestionnaire de bases de données. Par exemple, vous pouvez l'utiliser pour traduire le nom de l'application (Application Name).

Outbound Sequence No

Représente le numéro de séquence sortante. Il est utilisé pour mettre en corrélation des transactions sur différents systèmes.

Host Database Name

Le nom réel de la base de données à laquelle l'application est connectée. Dans le répertoire DCS, il s'agit du *nom de la base de données cible*.

Host Product ID

Le produit et la version qui fonctionnent sur le serveur. Il est indiqué sous la forme *PPPVVRRM*, où :

- PPP** Identifie le serveur de base de données grand système IBM (par exemple, DSN pour DB2 for z/OS, ARI pour DB2 Server for VSE & VM ou QSQ pour DB2 for IBM i)
- VV** Représente un numéro de version à deux chiffres, tel que 08.
- RR** Représente un numéro d'édition à deux chiffres, tel que 01.
- M** Représente un niveau de modification à un caractère (0-9 ou A-Z).

LIST DCS APPLICATIONS EXTENDED

Vous pouvez utiliser la commande LIST DCS APPLICATIONS avec l'option EXTENDED afin de générer un Rapport étendu. Le Rapport étendu affiche tous les champs répertoriés lorsque l'option SHOW DETAIL est spécifiée dans la commande, ainsi que les neuf champs suivants :

- Etat de l'application DCS
- Horodatage du changement d'état

- Plateforme client
- Protocole client
- ID de jeu de caractères codés de l'hôte (CCSID).
- ID de connexion client
- ID traitement de l'application client
- Alias de base de données utilisé au niveau de la passerelle
- Nom de la base de données DCS

Alors que les options de commande existantes répertorient les champs de manière horizontale, une ligne par application, la nouvelle option les répertorie de manière verticale, un champ par ligne.

La nouvelle syntaxe de la commande est la suivante :

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

L'exemple suivant illustre un exemple de résultats issus de cette commande, lorsque vous utilisez la nouvelle option EXTENDED :

Liste des applications DCS - Rapport étendu

```
ID de l'application client = 2002:91a:519:13:209:6bff:fe14:4fbb.8196.060306214224
Numéro de séquence          = 00001
ID autorisation             = NEWTON
ID autorisation sécurisé    =
Nom de l'application        = db2cli.exe
Descripteur de l'application = 37
Etat de l'application       = en attente de demande
Horodatage du changement d'état = Non disponible
Noeud client                = SAYYID
Niveau d'édition client     = SQL09000
Plateforme client           = NT
Protocole client            = TCP/IP
Page de codes client        = 1252
ID traitement de l'application client = 1192
ID de connexion client      = ISAYYID
ID application hôte         = G91A0D3A.P982.060306214231
Numéro de séquence         = 00001
Alias de base de données au niveau de la passerelle = MDB
Nom de la base de données DCS = MDB
Nom de la base de données hôte = MEXICO
Niveau d'édition hôte      = DSN08015
CCSID hôte                  = 1208
```

Le champ relatif à l'état de l'application contient l'une des trois valeurs suivantes :

1. CONNECT en attente - sortant. Signifie que la demande de connexion à une base de données grand système IBM a été émise et que DB2 Connect attend l'établissement de la connexion.
2. en attente de demande. Signifie que la connexion avec la base de données grand système IBM a été établie et que DB2 Connect attend une instruction SQL de l'application client.
3. en attente de réponse. Signifie que l'instruction SQL a été envoyée à la base de données grand système IBM.

Aussi, l'horodatage de la modification d'état s'affiche uniquement dans le rapport si l'unité d'oeuvre du moniteur système a été activée au cours du traitement. Autrement, l'état "Non disponible" s'affichera.

Moniteur d'état de santé et alertes

Le moniteur de santé DB2 for z/OS évalue régulièrement les règles de maintenance d'objet. S'il détermine qu'un objet a besoin d'une maintenance, il émet des alertes d'état. Vous pouvez afficher, lancer en exécution et sauvegarder les actions en réaction aux alertes d'état.

Moniteur de santé DB2 for z/OS - Généralités

Sur les systèmes z/OS, le moniteur de santé DB2 for z/OS est démarré en tant que tâche pour chaque sous-système DB2 à surveiller ou sur un membre dédié d'un groupe de partage de données.

Le moniteur de santé DB2 for z/OS déclenche l'évaluation des règles de maintenance des objets à des horaires et intervalles planifiés définis dans la règle. Les règles de maintenance d'objet sont créées à l'aide de l'assistant Création d'une règle de maintenance d'objets du Centre de contrôle DB2. Au cours de chaque évaluation, les critères de recommandation de maintenance sont comparés aux seuils définis dans la règle de maintenance d'objets pour déterminer la nécessité de procéder à une maintenance d'objets (autrement dit, COPY, REORG, RUNSTATS, STOSPACE, ALTER TABLESPACE ou ALTER INDEX sont requis) et pour identifier les états restreints, tels CHKP, sur l'espace de table, l'index et les objets de groupe de stockage. Lorsque les objets sont identifiés comme étant en état d'alerte au cours de l'évaluation de règles, les contacts pour alerte de santé sont avertis par messagerie électronique ou messenger de poche. La liste de ces contacts pour chaque sous-système DB2 est définie et gérée dans le Centre de contrôle.

Un instantané de la planification d'évaluation pour les règles, utilisé par le moniteur de santé pour déterminer le moment de déclenchement des évaluations de règle, est effectué par le moniteur de santé au démarrage. Cet instantané de planification est actualisé au démarrage du moniteur de santé ou lorsque le moniteur de santé reçoit une commande d'actualisation. Tout changement apporté à la planification d'évaluation d'une règle est pris en compte par le moniteur de santé lors de l'actualisation de la planification.

Le moniteur de santé est démarré et arrêté à partir de la console, à l'aide de commandes MVS système START et STOP, respectivement.

Une procédure cataloguée exemple (DSNHMONP) qui lance un moniteur de santé DB2, et une procédure cataloguée exemple (DSNHMONA) qui lance plusieurs moniteurs de santé DB2 dans un système MVS ou Parallel Sysplex, sont placées dans une bibliothèque de procédures par le travail d'installation DSNTIJHM.

Les vues, les tables, les ensembles de données, les procédures mémorisées, les fonctions définies par l'utilisateur et la table des ensembles des résultats qui sont utilisés par le moniteur de santé db2 ou les tâches connexes répertoriées ci-dessous sont créés et installés par les travaux d'installation DSNTIJCC et DSNTIJHM. DSNTIJCC et DSNTIJHM sont livrés avec FMIDs JDB771D et JDB881D.

Journal d'évaluation des règles

Les évaluations de règle déclenchées par le moniteur de santé DB2 sont consignées dans la table DSNACC.HM_EVAL_LOG. Une entrée est consignée lors du démarrage et de la fin d'une évaluation de règle. Les entrées de journal sont conservées pendant 7 jours. Elles sont ensuite supprimées de la table. La vue DB2 DSNACC.HM_ALERT_PO_EV, qui a été créée sur cette table par le travail d'installation DSNTIJCC, peut être utilisée pour afficher toutes les règles dont la dernière itération d'évaluation n'a pas abouti.

Démarrage, arrêt et régénération du moniteur de santé DB2 for z/OS

Sur le système z/OS, le moniteur de santé DB2 pour z/OS est démarré sous forme de tâche pour chaque sous-système DB2 à contrôler.

- Pour démarrer un moniteur de santé DB2, exécutez la commande système START MVS :

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

Les paramètres TRACE et REFRESH sont facultatifs.

nom_membre

Spécifie un membre de bibliothèque de procédure exécuté pour démarrer le moniteur de santé DB2, c'est-à-dire, DSNHMONP. Cette procédure cataloguée est créée par le travail d'installation DSNTIJHM.

ssid

Spécifie le nom ou l'identifiant du sous-système DB2 à surveiller.

trace

Spécifie l'indicateur de trace. Les valeurs admises sont les suivantes :

- ON - Activer la fonction de trace. Les enregistrements de trace sont inscrits dans SYSOUT
- OFF - Ne pas activer la fonction de trace

La valeur par défaut est OFF.

nn

Spécifie l'heure (au format 24 heures) à laquelle le moniteur de santé régénère l'image instantanée du calendrier d'évaluation pour déclencher des évaluations de règle. La valeur par défaut est 22.

- Pour démarrer plusieurs moniteurs de santé DB2, exécutez la commande système START MVS :

```
S membername
```

nom_membre

Un membre de bibliothèque de procédures exécuté pour démarrer plusieurs moniteurs de santé DB2, c'est à dire, DSNHMONA.

Remarque : Avant de démarrer plusieurs moniteurs de santé DB2 avec une commande START à l'aide de DSNHMONA, le fichier HMONPARM spécifié dans DSNHMONA proc doit être alimenté par la liste des sous-systèmes à surveiller. La procédure cataloguée et le fichier sont créés par le travail d'installation DSNTIJHM.

- Pour régénérer l'image instantanée du calendrier d'évaluation de règle utilisé par le moniteur de santé DB2 pour déterminer à quel moment les évaluations de règle doivent être déclenchées, exécutez la commande système suivante MODIFY MVS :

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

ssid

Nom ou identifiant du sous-système DB2 surveillé par le moniteur de santé DB2 que vous régénérez.

- Pour arrêter un moniteur de santé DB2, exécutez la commande système STOP MVS :

```
STOP HMONssid or P HMONssid
```

ssid

Nom ou identifiant du sous-système DB2 surveillé par le moniteur de santé DB2 que vous arrêtez.

Affichage, soumission et sauvegarde des actions recommandées

Pour afficher, soumettre et sauvegarder les actions recommandées pour les objets d'alerte identifiés au cours de l'évaluation de la règle, lancez la procédure DB2 mémorisée SYSPROC.DSNACCHR, créée par le travail d'installation DSNTIJCC. DSNACCHR est une procédure mémorisée qui détermine les actions recommandées pour les objets d'alerte identifiés au cours de l'évaluation de la règle et génère un travail JCL qui exécute des actions recommandées.

Le diagramme de syntaxe suivant illustre l'instruction SQL CALL permettant d'appeler DSNACCHR. La convention de liaison pour DSNACCHR étant GENERAL WITH NULLS, si vous transmettez des paramètres dans des variables d'hôtes, vous devez inclure un indicateur de valeur NULL avec chaque variable d'hôte. Les indicateurs de valeur NULL pour ces variables d'hôte d'entrée doivent être initialisés avant que vous puissiez exécuter l'instruction CALL.

Syntaxe

```
►—CALL DSNACCHR (—type-interrogation,—indicateur-santé,—id-règle,—tâche-ensemble,—►
  ►—fichier-nom,—membre-nom,—sauvegarde-option,—trace-indicateur,—►
    └─NULL┘ └─NULL┘ └─NULL┘
  ►—id-travail,—nom_travail,—jcl-temps-proc,—indic-trace,—Dernière-instruction,—►
  ►—code-retour,—msg-erreur )—►
```

type-interrogation

Précisez ce que vous comptez faire avec les actions recommandées pour les objets signalés en état d'alerte pendant l'évaluation de la règle. Les valeurs admises sont les suivantes :

- 0 - Visualiser les actions recommandées sur les objets en état d'alerte dans le cadre d'un travail JCL
- 1 - Soumettre un travail JCL exécutant les actions recommandées sur les objets en état d'alerte
- 2 - Soumettre le travail JCL exécutant les actions recommandées sur les objets en état d'alerte et placer le travail dans la file d'attente de stockage temporaire

- 3 - Sauvegarder les actions recommandées sur les objets en état d'alerte dans le cadre d'un travail JCL dans un membre de bibliothèque

type-interrogation est un paramètre d'entrée de type INTEGER.

indicateur-santé

Spécifie le type d'alerte que DSNACCHR place dans le travail JCL. Les valeurs admises sont les suivantes :

- RS - Etat restreint
- EX - Domaine dépassé
- RR - REORG requis
- CR - COPY requis
- RT - RUNSTATS requis
- SS - STOSPACE requis

indicateur-santé est un paramètre d'entrée du type VARCHAR(4).

id-règle

Spécifie une règle de maintenance d'objets. *id-règle* est un paramètre d'entrée du type VARCHAR(7).

tâche -ensemble

Spécifie l'ensemble de tâches d'une règle de maintenance d'objets ayant identifié les objets en état d'alerte que DSNACCHR place dans le travail JCL. Cet ensemble de tâches doit être identifié avec la règle et le type d'alerte spécifié dans les paramètres *id-règle* et *indicateur-santé*. *Tâche-ensemble* est un paramètre d'entrée du type INTEGER.

fichier-nom

Spécifie le nom complet qualifié d'un fichier partitionné (PDS) ou d'un fichier partitionné étendu (PDSE). Cette valeur doit être spécifiée si *type-interrogation* a la valeur 3. *fichier-nom* est un paramètre d'entrée de type VARCHAR(44).

nom-membre

Indique un membre de fichier partitionné (PDS) ou de fichier partitionné étendu (PDSE) spécifié dans le paramètre *fichier-nom* dans lequel le travail JCL de maintenance d'objet sera sauvegardé. Cette valeur doit être spécifiée si *type-interrogation* a la valeur 3. *nom-membre* est un paramètre d'entrée de type VARCHAR(8).

opt-sauvegarde

Indique comment sauvegarder le travail JCL de maintenance d'objet. Cette valeur doit être spécifiée si *type-interrogation* a la valeur 3. Les valeurs admises sont les suivantes :

- R - Remplacer
- A - Ajouter
- NM - Nouveau membre

opt-sauvegarder est un paramètre d'entrée de type VARCHAR(2).

Indicateur-trace

Indique si la fonction de trace est activée ou non. Les valeurs admises sont les suivantes :

- Y - Active la fonction de trace
- N - Désactive la fonction de trace

indicateur-trace est un paramètre d'entrée de type CHAR(1).

ID-travail

Lorsque *type-interrogation* a la valeur 1 ou 2, il indique l'ID travail d'un travail soumis. *id-travail* est un paramètre de sortie de type VARCHAR(8).

nom_travail

Lorsque *type-interrogation* a la valeur 1 ou 2, il indique le nom d'un travail soumis. *nom_travail* est un paramètre de sortie de type VARCHAR(8).

jcl-temps-proc

Indique l'heure à laquelle la demande a été traitée. *jcl-heure-traitement* est un paramètre de sortie de type TIMESTAMP.

dernière- instruction

Lorsque DSNACCHR renvoie une erreur grave (code de retour 12), cette zone contient l'instruction SQL qui était en cours d'exécution lorsque l'erreur s'est produite. *dernière-instruction* est un paramètre de sortie du type VARCHAR(2500).

code-retour

Code retour de l'exécution DSNACCHR. Les valeurs admises sont les suivantes :

- 0 - DSNACCHR exécuté correctement
- 12 - DSNACCHR terminé avec des erreurs graves. Le paramètre *msg-erreur* contient un message décrivant l'erreur. Le paramètre *dernier-instruction* contient l'instruction SQL qui était en cours d'exécution lorsque l'erreur s'est produite.

code-retour est un paramètre de sortie du type INTEGER.

msg-erreur

Lorsque DSNACCHR renvoie une erreur grave (code retour 12), cette zone contient des messages d'erreur, dont la mise en forme SQLCA. *msg-erreur* est un paramètre de sortie de type VARCHAR(1331).

DSNACCHR renvoie un ensemble de résultats lorsque le paramètre *type-interrogation* a la valeur 0. Le résultat contient le travail JCL généré par DSNACCHR. La table d'ensemble de résultats DSNACCHR est créée par le travail d'installation DSNTIJCC. Le tableau 12 affiche le format de l'ensemble de résultats.

Tableau 12. Format de l'ensemble de résultats DSNACCHR

Nom de colonne	Type de données	Description
JCLSEQNO	INTEGER	Numéro de séquence de la ligne de table (1,...,n)
JCLSTMT	VARCHAR(80)	Spécifie une instruction SQL

Affichage des récapitulatifs des alertes de santé

La fonction HEALTH_OVERVIEW fournit des informations issues du fichier récapitulatif des alertes de santé sous forme d'une table DB2. Ce fichier est créé par le travail d'installation DSNTIJHM.

Les données récapitulatives d'alerte de santé contiennent des informations concernant l'état du moniteur de santé DB2 et les statistiques du récapitulatif des alertes de chaque sous-système DB2 précédemment ou actuellement surveillé par le moniteur de santé sur ce système MVS ou Parallel Sysplex. Ces informations sont renvoyées au client avec une ligne pour chaque sous-système DB2 et recommandation d'alerte.

Le résultat de la fonction est une table DB2 avec les colonnes suivantes :

adresse-ip

Adresse IP du serveur DB2. C'est une colonne de type VARCHAR(40).

db2-ssid

Identifiant du sous-système DB2. Il s'agit d'une colonne de type VARCHAR(4).

indicateur-santé

Le type d'alerte. Les valeurs admises sont les suivantes :

- RS - Etat restreint
- EX - Domaine dépassé
- RR - REORG requis
- CR - COPY requis
- RT - RUNSTATS requis
- SS - STOSPACE requis
- PO - Echec d'évaluation de règle
- HM - Etat du moniteur de santé

indicateur-santé est une colonne du type VARCHAR(4).

nom-hôte

Le nom de domaine complet qualifié du serveur DB2. Il s'agit d'une colonne de type VARCHAR(255).

statistiques-récapitulatif

Etat du moniteur de santé DB2 si l'*indicateur-santé* est 'HM'. Les valeurs admises sont les suivantes :

- 0 Moniteur de santé non démarré
- 1 Moniteur de santé démarré
- -1 Etat du moniteur de santé inconnu

Sinon, le nombre total d'alerte d'objets avec le type d'alerte spécifié dans *indicateur-santé*. Il s'agit d'une colonne de type INTEGER.

état-alerte

Etat de l'alerte spécifiée dans *indicateur-santé*. Les valeurs admises sont les suivantes :

- 5 - Alarme
- 4 - Attention
- 3 - Avertissement
- 0 - Normal

état-alerte est toujours 0 lorsque *indicateur-santé* est 'HM'. Il s'agit d'une colonne de type INTEGER.

Le nom du programme externe de cette fonction est HEALTH_OVERVIEW, et le nom spécifique est DSNACC.DSNACCHO. Cette fonction est créée par le travail d'installation DSNTIJCC.

Exemple : Rechercher le nombre total des objets d'alerte pour lesquels COPY est requis pour le sous-système DB2 'ABCD' :

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

Affichage des objets alerte de santé

Les objets d'alerte identifiés au cours de la dernière itération réussie sont sauvegardés dans ces tables de référentiels d'objet d'alerte, en fonction de leur type d'objet.

Les objets d'alerte sont :

- DSNACC.HM_MAINT_TS pour les espaces table
- DSNACC.HM_MAINT_IX pour les index
- DSNACC.HM_MAINT_SG pour les groupes d'archivage

DB2 crée un nombre de vues de ces tables de référentiels d'objets d'alerte. Ces tables de référentiel de vues et d'objets d'alerte sont créés par le travail d'installation DSN TIJCC. Le tableau 13 répertorie les tables dans lesquelles chaque vue est définie ainsi que les descriptions des vues. Tous les noms de vue et de table sont caractérisés par le qualificatif DSNACC.

Tableau 13. Les vues des objets d'alerte de santé

Nom de la vue	Sur la table	Description de la vue
HM_ALERT_TS_RS	HM_MAINT_TS	Affiche tous les espaces de table à l'état restreint
HM_ALERT_TS_EX	HM_MAINT_TS	Affiche tous les espaces table dont les domaines ont dépassé une limite définie par l'utilisateur
HM_ALERT_TS_RR	HM_MAINT_TS	Affiche tous les espaces table qui requièrent REORG
HM_ALERT_TS_CR	HM_MAINT_TS	Affiche tous les espaces table qui requièrent COPY
HM_ALERT_TS_RT	HM_MAINT_TS	Affiche tous les espaces table qui requièrent RUNSTATS
HM_ALERT_IX_RS	HM_MAINT_IX	Affiche tous les index à l'état restreint
HM_ALERT_IX_EX	HM_MAINT_IX	Affiche tous les index dont les domaines ont dépassé une limite définie par l'utilisateur
HM_ALERT_IX_RR	HM_MAINT_IX	Affiche tous les espaces indexes qui requièrent REORG
HM_ALERT_IX_CR	HM_MAINT_IX	Affiche tous les index qui requièrent COPY
HM_ALERT_IX_RT	HM_MAINT_IX	Affiche tous les index qui requièrent RUNSTATS
HM_ALERT_SG_SS	HM_MAINT_SG	Affiche tous les groupes d'archivage qui requièrent STOSPACE

Partie 3. Haute disponibilité et DB2 Connect

Vous devez prendre en compte des considérations spécifiques liées à la haute disponibilité dans un environnement utilisant DB2 Connect. Si, pour une raison quelconque, un serveur de la base de données dans un réseau devient indisponible, il est important de pouvoir rediriger un poste de travail client vers un autre serveur de base de données du réseau.

Chapitre 11. Haute disponibilité et équilibrage de la charge de travail pour la connectivité de la base de données hôte

Sur le marché des technologies de l'information actuel, il existe une forte demande de disponibilité des données 24 heures sur 24. Cette demande doit être satisfaite afin qu'une entreprise puisse être compétitive et maintenir une croissance continue. De nos jours, de nombreuses applications Web, e-business et de feuille de calcul ont besoin d'accéder aux données de l'entreprise. Une connexion fiable, rapide et sécurisée aux bases de données grand système IBM doit être établie. Cette connexion doit être disponible en permanence et doit pouvoir gérer de fortes demandes de connexion dans des conditions de chargement critiques. Comment cette connexion peut-elle être générée ?

Scénario de haute disponibilité

Une entreprise possède divers postes de travail et serveurs d'applications fonctionnant sous Windows, Linux, et UNIX. Ces machines doivent pouvoir accéder aux données résidant sur plusieurs bases de données grand système IBM. Les applications fonctionnant sur ces machines ont besoin de connexions rapides et fiables aux bases de données. Tout le système est connecté au moyen d'un réseau Ethernet qui utilise le protocole TCP/IP.

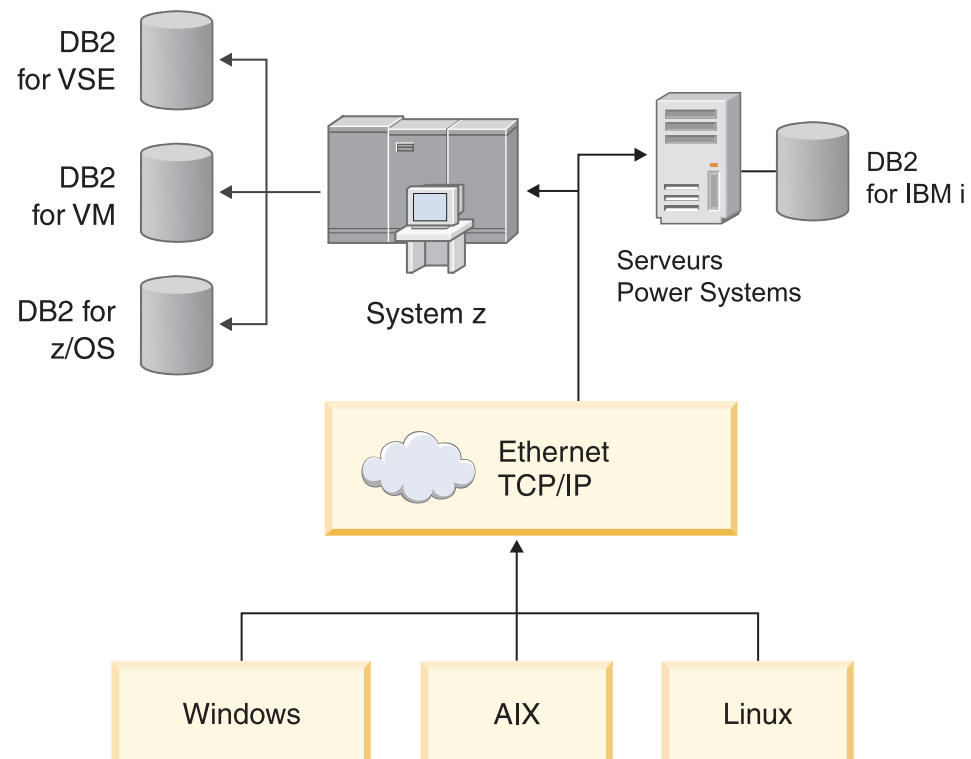


Figure 11. Exemple de scénario réseau

Pour que les postes de travail et les serveurs d'applications puissent accéder aux bases de données grand système IBM, un composant de connectivité doit servir d'intermédiaire. Ce composant doit offrir une connexion rapide, robuste et hautement disponible aux bases de données grand système IBM. Il doit également être évolutif afin d'anticiper toute croissance éventuelle du volume de connexions.

Consultez les liens connexes de cette rubrique pour afficher les détails concernant une solution utilisant DB2 Connect et la fonction de redirection automatique des clients.

Chapitre 12. Configuration et description de la redirection client automatique (DB2 Connect)

L'objectif principal de la fonction de redirection client automatique est de permettre la reprise d'une application IBM Data Server Client après un délai minimal en cas de perte de communication. Principe essentiel en termes de continuité des opérations, la redirection n'est toutefois possible que lorsqu'un autre emplacement est identifié pour la connexion client. Dans un environnement haute disponibilité non-DB2 Connect, la base de données accédée est généralement synchronisée entre le serveur DB2 d'origine et l'autre serveur DB2 via l'une des méthodes disponibles, comme par exemple l'utilisation d'un multiprocesseur de clusters à haute disponibilité (HACMP) ou la méthode HADR (reprise à haut niveau de disponibilité après incident).

Cependant, dans le cas du serveur DB2 Connect, puisque aucune synchronisation de bases de données locales n'est requise, il vous suffit de vous assurer que le serveur DB2 Connect initial et alternatif disposent d'une base de données grand système IBM cataloguée de sorte à être accessible à l'aide d'un alias de base de données identique.

Remarque : Dans un environnement de serveur DB2 Connect, il est possible de désigner un serveur DB2 Connect alternatif pour permettre une redirection automatique entre un client et le serveur DB2 Connect. Pour que la redirection puisse intervenir entre un produit DB2 Connect personnel ou serveur et un serveur de base de données grand système IBM, le serveur distant doit fournir une ou plusieurs adresses de substitution pour lui-même. Dans le cas de DB2 for z/OS, plusieurs adresses sont identifiées si la base de données représente un environnement de partage de données Sysplex.

Si le support Sysplex est activé, la fonction de redirection pour Sysplex peut être configurée entre DB2 Connect et le serveur de base de données hôte. La capacité de redirection pour Sysplex est une fonctionnalité de DB2 Connect qui permet à DB2 Connect de tenter une nouvelle connexion avec d'autres membres du groupe Sysplex en cas de perte de la communication avec le membre initial. Il n'est pas nécessaire que l'autre serveur soit catalogué dans le répertoire de la base de données pour que la fonction de redirection pour Sysplex puisse être activée sur DB2 Connect. Par défaut, cette fonction est activée dès lors que le support Sysplex est lui-même activé.

Afin que la reprise puisse être assurée pour un IBM Data Server Client après perte de communication avec un serveur DB2 Connect via la redirection client automatique, un autre emplacement de serveur DB2 Connect doit être indiqué avant que la perte de communication ne se produise. La commande UPDATE ALTERNATE SERVER FOR DATABASE permet de définir un emplacement alternatif de serveur DB2 Connect pour une base de données grand système IBM spécifique. Le nom d'hôte et le numéro de port de l'autre serveur sont indiqués en tant que partie intégrante de la commande. L'emplacement est stocké dans le fichier de répertoire de base de données système sur le serveur DB2 Connect. Afin de garantir que l'emplacement de l'autre serveur DB2 Connect indiqué s'applique à cette base de données pour tous les clients, cet emplacement doit être défini au niveau du serveur DB2 Connect. L'autre serveur est ignoré s'il est défini au niveau de l'instance client.

Par exemple, supposons qu'une base de données grand système IBM a été cataloguée à l'aide de l'alias de base de données db1 sur un serveur DB2 Connect S1 (avec le nom d'hôte db2conn1 et le numéro de port 122). L'administrateur de base de données souhaite spécifier un serveur DB2 Connect alternatif, S2, sur le nom d'hôte db2conn2 et le numéro de port 123. Voici la commande que l'administrateur de base de données devra exécuter sur le serveur DB2 Connect S1:

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

Après avoir spécifié l'emplacement du serveur alternatif DB2 Connect pour l'alias de base de données db1 sur le serveur DB2 Connect S1, ces informations sont renvoyées à IBM Data Server Client dans le cadre du processus de connexion. Si la communication entre IBM Data Server Client et le serveur DB2 Connect S1 est perdue pour une raison quelconque (généralement suite à une erreur de communication, telle qu'un code SQL -30081 ou -1224), IBM Data Server Client tentera de se reconnecter à db1, soit via le serveur DB2 Connect originel (S1), soit via le serveur alternatif DB2 Connect (S2), en alternant les tentatives entre ces deux serveurs. L'intervalle entre tentatives est initialement court puis s'allonge peu à peu.

Dès réussite d'une connexion, le code SQL -30108 est renvoyé afin d'indiquer que la connexion de base de données a été rétablie suite à l'erreur de communication. Le nom d'hôte ou l'adresse IP et le nom de service ou le numéro de port sont également renvoyés. S'il n'est pas possible de rétablir la communication client avec le serveur d'origine ou l'autre serveur, IBM Data Server Client ne renvoie à l'application que l'erreur de communication d'origine.

Il conviendra également de prendre en compte les remarques suivantes concernant la connectivité de l'autre serveur dans un environnement serveur DB2 Connect :

- Lors de l'utilisation d'un serveur DB2 Connect pour assurer l'accès à une base de données grand système IBM pour le compte de clients distants ou locaux, une confusion peut survenir concernant les informations de connectivité au serveur alternatif dans une entrée de répertoire de base de données système. Pour réduire ce risque de confusion, vous pouvez envisager de cataloguer deux entrées distinctes dans le répertoire de base de données système afin de représenter la même base de données grand système IBM. Cataloguez une entrée pour les clients distants et cataloguez une autre entrée pour les clients locaux.
- Toute information SYSPLEX renvoyée par un serveur DB2 for z/OS cible est conservée uniquement en cache sur le serveur DB2 Connect. Les informations d'un unique serveur de substitution sont écrites sur le disque. En présence de plusieurs serveurs de substitution ou de plusieurs serveurs actifs, les informations sont simplement maintenues en mémoire puis ces informations sont perdues une fois le processus arrivé à terme.

Chapitre 13. Configuration de la redirection automatique du client pour la technologie de distributeur de connexion client

Les technologies de distributeur de connexion client telles que WebSphere EdgeServer distribuent les demandes de reconnexion des applications client à un ensemble défini de systèmes en cas d'échec du serveur principal de base de données. Si vous utilisez cette technologie avec la redirection client automatique de DB2, vous devez identifier le distributeur lui-même comme serveur de remplacement auprès de la redirection client automatique de DB2.

Vous utilisez peut-être ma technologie de distributeur dans un environnement du type suivant :

Client —> technologie de distributeur > (DB2 Connect Server 1 ou DB2 Connect Server 2) —> DB2 z/OS

où :

- Le nom d'hôte TCP/IP du composant de technologie de distributeur est **DThostname**.
- Le nom d'hôte TCP/IP de DB2 Connect Server 1 est **GWYhostname1**.
- Le nom d'hôte TCP/IP de DB2 Connect Server 2 est **GWYhostname2**.
- Le nom d'hôte TCP/IP du serveur DB2 z/OS est **zOShostname**.

Le client est catalogué avec **DThostname** pour utiliser la technologie de distributeur afin d'accéder à l'un des serveurs DB2 Connect Servers. C'est la technologie de distributeur qui décide d'utiliser soit **GWYhostname1**, soit **GWYhostname2**. Une fois la décision prise, le client a une connexion socket directe à l'une de ces deux passerelles DB2 Connect. Une fois établie la connectivité socket au serveur DB2 Connect sélectionné vous disposez d'une connectivité normale de client vers un serveur DB2 Connect vers DB2 z/OS.

Par exemple, supposons que le distributeur choisisse **GWYhostname2**. Vous obtenez l'environnement suivant :

Client —> DB2 Connect Server 2 —> DB2 z/OS

Le distributeur ne réessaie aucune des connexions en cas d'incident de communication. Si vous souhaitez activer la fonction de redirection automatique du client pour une base de données dans cet environnement, le serveur de remplacement de la ou des bases de données associées sur le serveur DB2 Connect (DB2 Connect Server 1 ou DB2 Connect Server 2) doit être défini comme le distributeur (**DThostname**). Ensuite, en cas de blocage éventuel de DB2 Connect Server 1, la redirection automatique du client est déclenchée et une connexion client est relancée dans laquelle le distributeur est à la fois le serveur principal et le serveur de remplacement. Cette option vous permet de combiner et de conserver les fonctions du distributeur avec la fonction de redirection automatique du client DB2. Le fait de définir le serveur de remplacement comme un hôte différent du nom d'hôte du distributeur permet aux clients de continuer à utiliser la fonction de redirection automatique du client. Cependant, les clients établiront des connexions directes au serveur de remplacement défini et contourneront le distributeur, ce qui revient à l'annuler ainsi que tout l'intérêt qu'il apporte.

La redirection automatique du client intercepte les codes SQL suivants :

- sqlcode -20157
- sqlcode -1768 (code anomalie = 7)

Remarque : La redirection du client peut ne pas être informée à temps des échecs du socket si la valeur du paramètre "TCP Keepalive" des configurations du système d'exploitation est trop élevée. Le nom de ce paramètre de configuration varie selon la plateforme.

Partie 4. Réglage et DB2 Connect

Un environnement de base de données qui utilise DB2 Connect pour transférer des demandes et des réponses de la base de données entre les postes de travail client et les serveurs de la base de données pose des problèmes de performances particuliers. Il existe plusieurs méthodes permettant d'améliorer ou de préserver les performances dans cet environnement.

Chapitre 14. DB2 Connect remarques sur les performances

La *performance* est la façon dont un système informatique se comporte en fonction d'une charge de travail donnée. Elle est affectée par les ressources disponibles et la façon dont elles sont utilisées et partagées. Si vous souhaitez améliorer les performances, vous devez tout d'abord décider d'une définition du terme "performance".

Vous pouvez choisir divers *attributs de performances*, notamment :

Temps de réponse

L'intervalle entre le moment où l'application envoie la requête de base de données et le moment où l'application reçoit une réponse.

Débit des transactions

Le nombre d'unités d'oeuvre pouvant être traitées par unité de temps.
L'unité d'oeuvre peut être simple, comme l'extraction ou la mise à jour d'une ligne, ou compliquée, impliquant des centaines d'instructions SQL.

Vitesse de transfert des données

Le nombre d'octets de données transférés entre l'application DB2 Connect et la base de données grand système IBM par unité de temps.

Les performances seront limitées par les ressources matérielles et logicielles disponibles. L'unité centrale, l'espace mémoire et les adaptateurs réseau sont des exemples de ressources matérielles. Les sous-systèmes de communication, les sous-systèmes de pagination, mbuf pour AIX, sont des exemples de ressources logicielles.

Flots de données

La figure 12 illustre le chemin des données circulant entre le serveur de base de données grand système IBM et le poste de travail via DB2 Connect.

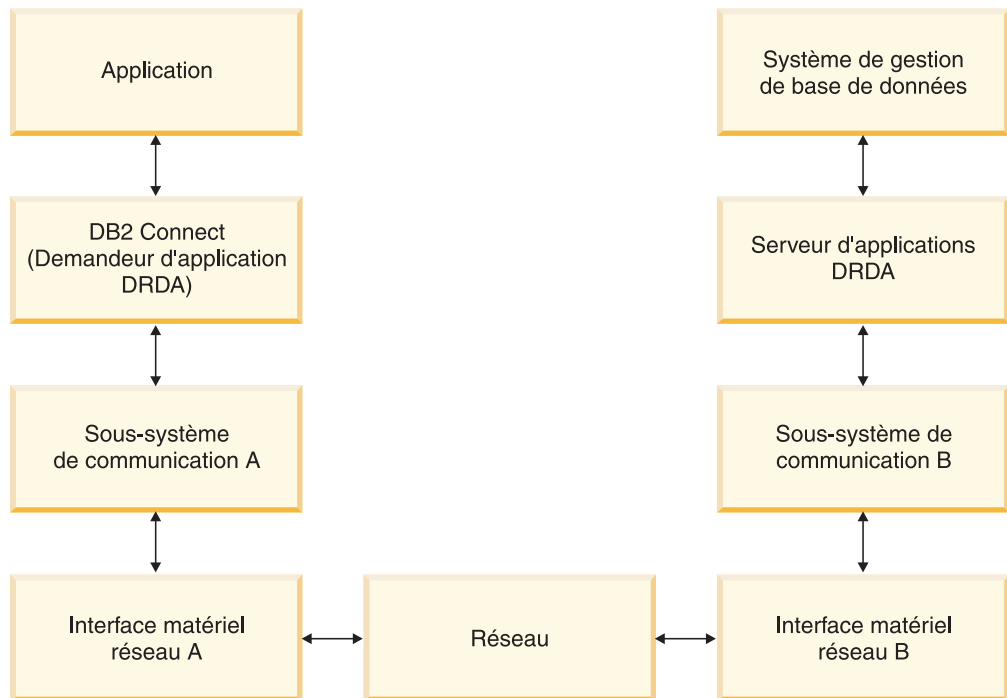


Figure 12. Flots de données dans DB2 Connect

- La base de données grand système IBM ainsi qu'une partie du sous-système de communication B s'exécutent généralement sur le même système. Ce système se compose d'une ou plusieurs unités centrales, d'une mémoire système, d'un sous-système E-S, d'une unité de stockage à accès direct et d'un système d'exploitation. Comme d'autres programmes peuvent partager ces composants, des conflits de ressources peuvent engendrer des problèmes de performance.
- Le réseau se compose d'une combinaison de câbles, de concentrateurs, de lignes de communication, de commutateurs et autres contrôleurs de communication. Par exemple, l'interface matérielle réseau peut être constituée de contrôleurs de communication, comme le modèle 3745 ou 3172, ou d'un adaptateur Token Ring pour un serveur IBM Power Systems. Plusieurs supports de transmission peuvent être impliqués entre les interfaces matérielles réseau A et B.
- L'interface matérielle réseau A peut être une carte en anneau à jeton, une carte Ethernet** ou LAN, ou une carte prenant en charge les protocoles SDLC ou X.25.
- DB2 Connect et le sous-système de communication A sont généralement situés sur le même système. Dans cette discussion, il est supposé que l'application se trouve également sur le même système.

Goulots d'étranglement

Le débit des transactions dépend du composant le plus lent du système. Si vous identifiez un goulot d'étranglement des performances, vous pouvez atténuer l'incident en modifiant les paramètres de configuration, en allouant davantage de ressources au composant concerné, en mettant à jour le composant ou en ajoutant un nouveau composant afin de décharger une partie du travail.

Vous pouvez utiliser divers outils afin de déterminer le temps qu'une requête passe dans chaque composant. Cela vous donnera une idée des composants nécessitant un réglage ou une mise à niveau pour améliorer les performances. Par exemple, si vous déterminez qu'une requête passe 60 % du temps dans la machine DB2 Connect, vous souhaitez peut-être régler DB2 Connect ou (si vous possédez des clients distants) ajouter une autre machine DB2 Connect sur le réseau.

Conduite de tests de performances

Les *tests de performances* comparent les performances d'un environnement avec celles d'un autre environnement. Les tests de performances peuvent débuter en exécutant l'application de test dans un environnement standard. Les problèmes de performances étant restreints, des scénarios de test spécialisés peuvent être développés pour limiter la portée de la fonction testée et observée.

Les tests de performances ne doivent pas être complexes. Les scénarios de test spécialisés n'ont pas besoin d'émuler une application complète pour obtenir des informations utiles. Commencez avec des mesures simples et augmentez uniquement la complexité quand cela est justifié.

Caractéristiques de bons tests de performances :

- Chaque test est répétitif.
- Chaque itération d'un test a lieu dans le même état système.
- Le matériel et le logiciel utilisés pour les tests de performances sont compatibles avec votre environnement de production.
- Aucune autre fonction ou application n'est active dans le système à l'exception de celles qui ont été mesurées sauf si le scénario inclut d'autres activités exécutées sur le système.

Remarque : Les applications démarrées utilisent de l'espace mémoire même lorsqu'elles sont réduites ou en veille. Elles peuvent donner lieu à de la pagination ou biaiser les résultats des tests de performances.

Outils de performance

Les tableaux suivants répertorient certains outils qui peuvent vous aider à mesurer les performances système. Puisque ces outils utilisent eux-mêmes les ressources système, vous souhaitez peut-être ne pas les activer constamment.

Tableau 14. Outils de performance pour l'utilisation de l'unité centrale et de l'espace mémoire

Système	Outil	Description
AIX	vmstat, time, ps, tprof	Fournissent des informations sur les conflits de l'unité centrale et de l'espace mémoire sur le poste de travail DB2 Connect et les clients distants.
HP-UX	vmstat, time, ps, monitor etglance si disponible	
Windows	Moniteur de performances Microsoft	

Tableau 15. Outils de performance pour l'activité de la base de données

Système	Outil	Description
Tous systèmes	Moniteur de base de données	Détermine si l'incident provient de la base de données.
System z	IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS, ASG-TMON for DB2 (ASG), and CA Insight Performance Monitor for DB2 for z/OS (Computer Associates International, Inc.)	
Windows	Moniteur de performances Microsoft	

Tableau 16. Outils de performance pour l'activité réseau

Système	Outil	Description
AIX	netpmon	Fait état des statistiques réseau de faible niveau, notamment les statistiques TCP/IP telles que le nombre de paquets ou de trames reçu(e)s par seconde.
Contrôleur réseau tel que le 3745	Moniteur de performances NetView	Fait état de l'utilisation de contrôle de transmission des données et VTAM.
Linux et UNIX	netstat	Gère le trafic TCP/IP.

Chapitre 15. Optimisation de l'accès ODBC

La base de données DB2 offre l'optimisation spécifique conçue pour améliorer les performances de communication via la connectivité ODBC. Ces améliorations sont disponibles pour Microsoft Access, Lotus Approach ou Visual Basic. Vous bénéficiez ainsi d'un rendement ODBC plus rapide grâce au programme d'aide à la configuration de DB2.

Pour activer la connectivité ODBC optimisée :

- Si vous définissez une nouvelle connexion :
 1. Démarrez le programme d'aide à la configuration de DB2.
 2. Ouvrez le menu Objets sélectionnés et sélectionnez Ajout d'une base de données avec l'assistant...
 3. Suivez les pages de l'assistant jusqu'à ce vous parveniez à la page **Source de données**.
 4. Vérifiez **Enregistrement de la base de données pour CLI/ODBC**.
 5. Spécifiez comment les applications CLI/ODBC accédant à cette base de données doivent être enregistrées :
 - **Comme source de données système** signifie que la base de données est disponible pour tous les utilisateurs du système.
 - **Comme source de données utilisateur** signifie que vous êtes l'unique utilisateur à pouvoir accéder à la base de données.
 - **Comme source de données fichier** signifie qu'un fichier contenant des informations sur les sources de données sera créé. Ce fichier de source de données peut être partagé avec d'autres postes de travail si vous possédez une connexion TCP/IP. Autrement, le fichier peut uniquement être utilisé par cet ordinateur
 6. Entrez le **Nom de la source de données**.
 7. (Facultatif) Sélectionnez une application dans la liste **Optimisation pour l'application** afin d'optimiser les paramètres de source de données d'une application donnée.
 8. Cliquez sur **OK** et quittez l'Assistant de configuration.
- Si vous procédez à la mise à jour d'une connexion existante :
 1. Démarrez le programme d'aide à la configuration de DB2.
 2. Cliquez deux fois sur l'alias de la base de données que vous souhaitez optimiser.
 3. Cliquez sur **Source de données**.
 4. Vérifiez **Enregistrement de la base de données pour CLI/ODBC**.

5. Spécifiez comment les applications CLI/ODBC accédant à cette base de données doivent être enregistrées :
 - **Comme source de données système** signifie que la base de données est disponible pour tous les utilisateurs du système.
 - **Comme source de données utilisateur** signifie que vous êtes l'unique utilisateur à pouvoir accéder à la base de données.
 - **Comme source de données fichier** signifie qu'un fichier contenant des informations sur les sources de données sera créé. Ce fichier de source de données peut être partagé avec d'autres postes de travail si vous possédez une connexion TCP/IP. Autrement, le fichier peut uniquement être utilisé par cet ordinateur
6. Entrez le **Nom de la source de données**.
7. (Facultatif) Sélectionnez une application dans la liste **Optimisation pour l'application** afin d'optimiser les paramètres de source de données d'une application donnée.
8. Cliquez sur **OK** et quittez l'Assistant de configuration.

Chapitre 16. Conception d'application

Lorsque vous créez une application, vous pouvez améliorer les performances de diverses manières.

SQL composé et procédures mémorisées

Pour les applications qui envoient et reçoivent de nombreuses commandes et réponses, le temps système réseau peut être conséquent. Le SQL composé et les procédures mémorisées constituent deux façons de réduire ce temps système.

Si une application envoie plusieurs instructions SQL sans intervenir dans la logique de programmation, vous pouvez utiliser le SQL composé. Si une logique de programmation est requise au sein du groupe ou des instructions SQL, vous pouvez utiliser les procédures mémorisées.

Toutes les instructions exécutables à l'exception des instructions suivantes, peuvent être incluses dans une instruction SQL composée :

- CALL
- FETCH
- CLOSE
- OPEN
- SQL composé
- Connect
- Prepare
- Release
- Describe
- Rollback
- Disconnect
- Set connection
- execute immediate

Les procédures mémorisées aident à réduire le trafic réseau en plaçant la logique du programme au niveau du serveur. Vous pouvez valider automatiquement lorsque vous quittez la procédure. Vous pouvez également renvoyer des ensembles de résultats qui réduisent la logique applicative au niveau du client.

Groupement des requêtes

Le groupement des requêtes de bases de données associées (instructions SQL) dans une requête de base de données peut réduire le nombre de requêtes et de réponses transmises à travers le réseau.

Par exemple, le groupement des instructions suivantes :

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

dans

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

permet d'envoyer moins de requêtes à travers le réseau.

Vous pouvez également utiliser des mots clés tels que IN et BETWEEN afin de diminuer le nombre de lignes renvoyées. En outre, vous pouvez utiliser les mots clés WHERE, IN et BETWEEN dans les instructions UPDATE et DELETE.

Logique des prédicats

Vous pouvez utiliser la logique des prédicats pour demander uniquement les lignes et colonnes nécessaires. Le trafic réseau et le temps système de l'unité centrale sont ainsi réduits pour la transmission de données.

Par exemple, n'utilisez pas la requête :

```
SELECT * FROM TABLEA
```

si seule la première ligne de TABLEA avec le ROW_ID=1 est réellement nécessaire ou si seules les colonnes 1 et 2 sont nécessaires.

Blocage de données

Utilisez le blocage de données si vous vous attendez à recevoir de grandes quantités de données du serveur. Le blocage améliore l'utilisation de la bande passante réseau et diminue la charge sur l'UC du serveur de base de données grand système IBM et sur celle du serveur DB2 Connect. Une quantité fixe de temps système de l'unité centrale et du réseau est attribuée à chaque message envoyé ou reçu quelle que soit sa taille. Le blocage de données réduit le nombre de messages requis pour la même quantité de transfert de données.

Grâce au blocage, la première ligne de données d'une requête ne sera pas livrée à l'application avant que le premier bloc ne soit reçu. Le blocage augmente le délai d'extraction de la première ligne mais améliore le délai d'extraction des lignes suivantes.

Un autre point est la quantité d'espace mémoire utilisée. La partie active de l'espace mémoire augmente lorsque le blocage est activé.

Dans DB2 Connect, vous pouvez contrôler la quantité de données transférée au sein de chaque bloc.

Pour appeler le blocage, utilisez l'option BLOCKING de la commande prep ou bind. Le blocage est activé si :

- Le curseur est en lecture seulement, ou
- Le curseur est équivoque et que le blocage est spécifié dans la commande prep ou bind.

Remarque : Lorsque vous utilisez le SQL dynamique, le curseur est équivoque.

Instructions SQL avec BLOCKING

Les instructions SQL SELECT actualisables (à l'aide des instructions UPDATE/DELETE WHERE CURRENT OF) sont des requêtes non bloquantes ; aussi, ne les utilisez que lorsqu'elles sont absolument nécessaires.

Une instruction SELECT actualisable garantit que la ligne ne sera pas modifiée entre le moment où l'instruction SELECT est achevée et le moment où l'instruction UPDATE/DELETE est exécutée. Si ce niveau d'accès concurrent n'est pas important pour votre application, une alternative consiste à utiliser l'instruction DELETE ou UPDATE avec des critères de recherche basés sur des valeurs renvoyées par une instruction SELECT non actualisable.

Pour une instruction SELECT en lecture seulement, spécifiez FOR FETCH ONLY, sauf sous VM et VSE, sur lesquels l'instruction n'est pas prise en charge.

SQL statique et dynamique

Utilisez le SQL statique autant que possible. Il vous évite la préparation de la section d'exécution SQL ainsi que l'utilisation de curseurs équivoques. Si vous ne pouvez éviter d'utiliser le SQL dynamique, vous pouvez procéder comme suit pour réduire le trafic réseau et améliorer les performances :

- Si l'instruction est une instruction SELECT qui doit être préparée, exécutez `PREPARE ... INTO SQLDA`. La structure SQLDA doit être allouée à la taille maximale requise par vos paramètres. Si le nombre maximal de colonnes est x et que le nombre n'est pas supposé diminuer, allouez une structure SQLDA avec x SQLVAR. Si le nombre de colonnes potentielles est incertain (et que l'espace mémoire n'est pas un problème), utilisez le nombre maximal de SQLVAR (256).

Si l'allocation de la structure SQLDA n'est pas suffisante pour stocker la structure SQLDA retour, le programme doit exécuter une autre instruction DESCRIBE avec une structure SQLDA suffisante pour stocker à nouveau les résultats. Le trafic réseau s'en trouverait augmenter.

N'utilisez pas la séquence PREPARE et DESCRIBE. L'utilisation de l'instruction `PREPARE.....INTO` engendre de meilleures performances.

- Exécutez les instructions SQL COMMIT ou ROLLBACK liées statiquement au lieu des instructions COMMIT ou ROLLBACK.
- S'il ne s'agit pas d'une instruction SELECT, COMMIT ou ROLLBACK, exécutez l'instruction EXECUTE IMMEDIATE pour exécuter l'instruction au lieu de la séquence PREPARE et EXECUTE.
- Les applications ODBC utilisent le SQL dynamique. Vous pouvez utiliser la fonction de profilage statique CLI/ODBC pour améliorer les performances. Cette fonction vous permet de capturer et de convertir des appels ODBC dans des instructions statiques stockées dans un module de base de données. Les performances que vous obtiendrez dépendent de la complexité de votre application.

Autres remarques concernant le langage SQL

L'utilisation de l'Interpréteur de commandes (CLP) est généralement plus lente que l'intégration de SQL dynamique dans le programme car l'interpréteur de commandes doit analyser l'entrée avant d'envoyer le SQL dans le moteur de base de données. L'interpréteur de commandes formate également des données reçues qui ne sont peut-être pas nécessaires pour votre application.

Les instructions SQL dans un langage interprété, tel que REXX, sont sensiblement plus lentes que dans un langage compilé, tel que C.

Il existe deux types d'instruction CONNECT, appelés type 1 et type 2. Avec l'instruction Connect de type 2, la connexion à la base de données place la connexion précédente dans un état de veille sans la supprimer. Si vous basculez ultérieurement vers une connexion en veille, vous évitez ainsi le temps système lié au chargement des bibliothèques et à la configuration des structures de données internes. Aussi, l'utilisation de l'instruction Connect de type 2 peut améliorer les performances des applications accédant à plusieurs bases de données.

Chapitre 17. Gestion des connexions

La gestion des connexions comprend deux éléments : le regroupement de connexions et le concentrateur de connexion. Le regroupement de connexions réduit le temps système des connexions à la base de données et gère le volume des connexions. Le concentrateur de connexion augmente l'évolutivité dans votre environnement de travail en optimisant les ressources utilisées par les serveurs de la base de données. Ces deux éléments sont décrits ici.

Regroupement de connexions

Les serveurs DB2 Connect, tels que DB2 Connect Enterprise Edition, offrent généralement des connexions à la base de données à des milliers de requêtes client simultanées. L'établissement et la fermeture de connexions au serveur de base de données peut être un processus consommant énormément de ressources affectant à la fois les performances du serveur de base de données et du serveur DB2 Connect.

Ce problème est particulièrement évident dans des environnements Web dans lesquels chaque visite sur une page Web peut requérir la génération d'une nouvelle connexion au serveur de base de données, la réalisation d'une requête et la fermeture d'une connexion. Afin de réduire le temps système, le serveur DB2 Connect utilise le regroupement de connexions afin de gérer les connexions ouvertes à la base de données dans un pool facile d'accès.

La plupart des applications basées sur les technologies Web exécutent de grands volumes de brèves transactions. Une transaction Web typique est exécutée en tant que composant de sa propre connexion. En d'autres termes, l'exécution d'une transaction signifie l'établissement d'une connexion à la base de données et la fermeture de cette connexion après quelques instructions SQL. Ce processus d'établissement et de fermeture de connexion est très onéreux. Il implique la création d'un agent DB2 Connect chargé d'établir une connexion réseau entre cet agent et le serveur DB2 ainsi que la création d'une unité d'exécution DB2 sur le serveur. Pour les connexions à exécution plus longue, ces coûts sont amortis sur toutes les transactions exécutées à l'aide de cette connexion. Mais, en règle générale, pour une transaction Web typique, ces coûts excèdent souvent le coût d'exécution de la transaction à proprement parler.

Le regroupement de connexions est une technique qui permet de réutiliser une infrastructure de connexion établie pour des connexions ultérieures. Lorsqu'une instance DB2 Connect démarre, un regroupement d'agents de coordination est créé. Lorsqu'une demande de connexion arrive, un agent est affecté à cette requête. L'agent se connecte au serveur DB2 et une unité d'exécution est créée dans DB2. Lorsque l'application émet une demande de déconnexion, l'agent ne transmet pas cette demande au serveur DB2. Au lieu de cela, l'agent est replacé dans le regroupement. L'agent placé dans le regroupement possède toujours sa connexion au serveur DB2 et à l'unité d'exécution DB2 correspondante. Lorsqu'une autre application émet une demande de connexion, cet agent est affecté à cette nouvelle application. Pour garantir la sûreté de cette opération, les informations relatives à l'identité de l'utilisateur sont transmises à l'unité d'exécution DB2 qui procède à l'authentification de l'utilisateur.

Le regroupement de connexions de DB2 améliore les performances de manière considérable dans de tels environnements. DB2 Connect gère les connexions ouvertes à la base de données dans un regroupement disponible. Lorsqu'un client demande une connexion, elle peut être fournie à partir de ce regroupement de connexions déjà établies. Le regroupement de connexions réduit de manière significative le temps système généralement dépensé dans l'ouverture et la fermeture de ces connexions.

Le regroupement de connexions est un procédé transparent pour les applications qui se connectent à l'hôte via DB2 Connect. Lorsqu'une application demande à se déconnecter de l'hôte, DB2 Connect supprime la connexion entrante à l'application mais conserve la connexion sortante vers l'hôte dans un regroupement. Lorsqu'une nouvelle application demande une connexion, DB2 Connect utilise une connexion du regroupement existant. L'utilisation de connexions déjà établies réduit le temps de connexion global ainsi que les coûts de connexion élevés de l'unité centrale sur l'hôte.

Les agents DB2 Connect peuvent posséder deux états : en veille et actif. Un agent est actif lorsqu'il exécute un travail pour une application. Une fois ce travail terminé, l'agent passe à l'état de veille et attend la soumission d'un nouveau travail provenant de la même ou d'une autre application. Tout agent mis en veille est conservé dans un regroupement d'agents en veille. Vous pouvez configurer la taille de ce regroupement à l'aide du paramètre de configuration **num_poolagents**. Ce paramètre équivaut au nombre maximal d'agents en veille que vous souhaitez que le système gère. La définition de ce paramètre sur 0 revient à désactiver la fonction de regroupement de connexions. Par défaut, ce paramètre est défini sur AUTOMATIC, avec une valeur de 100. Du fait de l'option AUTOMATIC, DB2 Connect gère automatiquement le nombre d'agents en veille dans leur pool.

DB2 Connect n'établit aucune connexion à la base de données avant de recevoir sa première demande client. Toutefois, vous pouvez remplir le regroupement d'agents en veille avant qu'un client n'effectue une demande. Le regroupement peut être rempli au démarrage à l'aide du paramètre de configuration **num_initagents**. Ce paramètre détermine le nombre d'agents en veille à créer au démarrage. Au commencement, ces agents en veille ne posséderont pas de connexions vers le serveur de base de données hôte.

Lorsqu'un client demande une connexion à l'hôte, DB2 Connect tente d'obtenir un agent parmi les agents placés dans le regroupement possédant une connexion établie vers le serveur de base de données hôte. S'il échoue, il tente de trouver un agent disponible dans le regroupement en veille. Si le regroupement est vide, DB2 Connect crée un nouvel agent.

Vous pouvez contrôler le nombre d'agents pouvant être actifs simultanément à l'aide du paramètre de configuration **max_coordagents**. Une fois ce nombre atteint, les nouvelles connexions échoueront avec le code d'erreur sqlcode SQL1226. (Ce code signifie que le nombre maximal de connexions concurrentes sortantes a été dépassé.) Par défaut, ce paramètre est défini sur AUTOMATIC, avec une valeur de 200. Du fait de l'option AUTOMATIC, DB2 Connect gère automatiquement le nombre d'agents de coordination.

La variable de registre DB2 DB2CONNECT_IN_APP_PROCESS permet aux applications fonctionnant sur la même machine que le serveur DB2 Connect d'exécuter DB2 Connect au sein des processus applicatifs (le comportement par défaut) ou de connecter l'application au serveur DB2 Connect et d'établir une connexion à l'hôte qui sera exécutée au sein d'un agent. Pour une application qui

utilise le regroupement de connexions, les connexions vers l'hôte doivent être établies à partir des mêmes agents de serveur DB2 Connect ; par conséquent, DB2CONNECT_IN_APP_PROCESS doit être défini sur NO.

Regroupement de connexions DB2 Connect versus regroupement de connexions Application Server

Le regroupement de connexions est un impératif pour n'importe quelle application fondée sur les technologies Web devant prendre en charge de grands volumes de transactions. La plupart des serveurs d'applications Web fournissent désormais leur propre regroupement de connexions à la base de données. Par exemple, Microsoft MTS (COM+) et IBM WebSphere offre le regroupement de connexions.

Les mécanismes de regroupement des applications mis en oeuvre par ces serveurs diffèrent énormément de la solution proposée par les serveurs DB2 Connect. Puisque les serveurs d'applications regroupent les connexions pour leur propre utilisation, ils supposent généralement que l'ID utilisateur, le mot de passe, les niveaux d'isolement, etc, sont exactement les mêmes pour toutes les connexions. Qui plus est, les serveurs d'applications regroupent uniquement les connexions établies par le même processus. Cela signifie que les connexions issues d'autres machines, utilisateurs ou processus ne sont pas regroupées. Alors que les techniques de regroupement de ces serveurs d'applications sont efficaces lors de la réutilisation de connexions établies par la même instance d'une application, elles sont absolument inefficaces lorsqu'il s'agit de regrouper des connexions provenant d'utilisateurs, de serveurs, etc, divers.

Le regroupement de connexions proposé par les serveurs DB2 Connect est complètement indépendant des applications, des machines et des utilisateurs à partir desquels ces connexions ont été établies. Les connexions établies à partir de clients et de serveurs d'applications multiples, tous possédant des ID utilisateur différents, peuvent réutiliser n'importe quelle connexion, ce qui engendre une meilleure utilisation des ressources regroupées.

Quel est le bon type de regroupement de connexions à utiliser ? Les deux. En règle générale, l'utilisation conjointe des regroupements de connexions DB2 Connect et Application Server est une bonne stratégie car les deux types de regroupement n'interfèrent pas l'un avec l'autre. Même lorsque le regroupement de connexions Application Server est activé, le regroupement de connexions DB2 Connect permet la réutilisation de connexions par divers serveurs d'applications ainsi que par d'autres clients utilisant le serveur DB2 Connect.

Concentrateur de connexion

Le concentrateur de connexion réduit le nombre de ressources requises sur les serveurs de base de données DB2 for z/OS afin de prendre en charge un grand nombre de postes de travail et d'utilisateurs Web. Cette fonction peut accroître de manière très significative l'extensibilité de votre solution DB2 for z/OS et DB2 Connect tout en assurant une tolérance des pannes et un équilibrage de charge des transactions dans les environnements DB2 for z/OS avec partage de données.

Le concentrateur de connexion permet aux applications de rester connectées sans qu'aucune ressource ne soit consommée sur le serveur hôte DB2. Vous pouvez comptabiliser des milliers d'utilisateurs actifs dans les applications alors que seules quelques unités d'exécution sont actives sur le serveur hôte DB2.

La technologie de *concentrateur de connexion* de DB2 Connect permet aux serveurs DB2 Connect, tels que DB2 Connect Enterprise Edition, de prendre en charge des milliers d'utilisateurs exécutant simultanément des transactions commerciales, tout en réduisant considérablement les ressources requises sur les serveurs de base de données hôte System z ou IBM Power Systems. Pour atteindre cet objectif, cette technologie concentre les charges de travail de toutes les applications sur un nombre beaucoup plus réduit de connexions au serveur de base de données hôte System z ou IBM Power Systems. Bien que cette fonction puisse sembler similaire à la fonction de regroupement de connexions décrite ci-avant, il s'agit d'une approche plus sophistiquée pour réduire la consommation des ressources par les applications de traitement de transactions en ligne très volumineuses.

Le concentrateur de connexion prend le concept d'un agent et le divise en deux entités :

- L'agent logique qui représente la connexion d'une application.
- L'agent de coordination qui possède la connexion et l'unité d'exécution DB2 et exécute les requêtes des applications.

Lorsqu'une nouvelle application tente de se connecter à un hôte, elle se voit attribuer un agent logique. Pour transmettre le SQL à la base de données, l'agent de coordination est nécessaire dès qu'une nouvelle transaction démarre. La clé de cette architecture est le fait que l'agent de coordination est :

- Dissocié de l'agent logique
- Renvoyé par le regroupement lorsqu'une transaction s'achève suite à la validation ou l'annulation

Une autre fonction clé est la méthode d'affectation des agents de coordination à de nouvelles transactions dans un environnement de partage de données. DB2 Connect implémente un algorithme d'ordonnement sophistiqué qui utilise les informations du Work Load Manager (WLM) de System z. Ces informations sont utilisées afin de répartir la charge de travail entre les membres d'un groupe de partage de données conformément aux critères configurés dans le WLM. Le WLM connaît la charge de chaque membre mais également leur disponibilité. Ainsi DB2 Connect peut de manière transparente attribuer les tâches des membres surchargés ou pour lesquels une erreur est survenue à des membres actifs sous-utilisés. Le concentrateur de connexion DB2 Connect est activé lorsque le nombre maximal d'agents logiques (*max_connections*) défini est supérieur au nombre d'agents de coordination (*max_coordagents*).

Le regroupement de connexions vous permet d'épargner les coûts d'établissement d'une connexion lorsqu'une application qui s'achève n'a plus besoin de sa connexion. En d'autres termes, une application doit se déconnecter avant qu'une autre application ne puisse réutiliser une connexion répartie.

Le concentrateur de connexion permet également à DB2 Connect de mettre une connexion à la disposition d'une application lorsqu'une autre application a achevé sa transaction sans avoir besoin que cette autre application ne se déconnecte. Une connexion de serveur de base de données et ses hôtes et ressources DB2 Connect associés sont essentiellement utilisés par une application uniquement lorsqu'elle dispose d'une transaction active. Une fois la transaction achevée, la connexion et les ressources associées sont disponibles pour une autre application qui doit exécuter une transaction.

Dans les versions précédentes de DB2 Connect, toute application active possédait une EDU (Engine Dispatchable Unit) qui gérait la connexion à la base de données ainsi que d'autres requêtes d'application. Cette EDU était généralement appelée

agent de coordination. Chaque agent de coordination recherche l'état ou le contexte de l'application et de l'EDU. Chaque EDU consomme une quantité considérable d'espace mémoire lorsque le nombre de connexions augmente et le changement de contexte entre les agents engendre une utilisation supplémentaire du temps système.

Dans l'architecture susmentionnée, il existe une relation biunivoque entre les connexions et les EDU. Le concentrateur de connexion, cependant, offre une relation à origines multiples et destination unique entre les connexions et les EDU. La relation entre les connexions (X) et les EDU (Y) est dorénavant $X \geq Y$.

Le concentrateur de connexion divise l'agent en deux entités, un *agent logique* et un *agent exécutant*. Les agents logiques représentent une application, sans référencer une EDU particulière. L'agent logique contient toutes les informations et tous les blocs de contrôle requis par une application. Si n applications sont connectées au serveur, n agents logiques seront présents sur le serveur. Les agents exécutants sont des EDU physiques qui exécutent les requêtes des applications sans posséder de connexion particulière avec une application spécifique. Les agents exécutants s'associent aux agents logiques pour effectuer des transactions. Une fois la transaction terminée, ils interrompent cette association et retournent dans le regroupement disponible.

Une entité connue sous le nom de *répartiteur* affecte des agents exécutants à des agents logiques. Les limitations du nombre de descripteurs de fichiers ouverts définies sur certaines plateformes informatiques peuvent engendrer la présence de plusieurs instances de planification.

Restrictions du concentrateur de connexion

Il existe plusieurs restrictions importantes relatives à l'utilisation du concentrateur de connexion DB2 Connect. Lisez les informations suivantes dans leur intégralité avant d'essayer d'utiliser le concentrateur de connexion sur votre système.

Restrictions générales :

- Le concentrateur repose sur le protocole TCP/IP pour établir des connexions entrantes à partir de clients locaux et distants. Seules les connexions entrantes utilisant le protocole TCP/IP ou locales (communication interprocessus) pourront bénéficier des connexions sortantes réparties. Le concentrateur accepte les connexions établies via d'autres protocoles de communication, tels que les tubes nommés, mais n'autorisera pas ces connexions à utiliser les fonctions de concentration XA.
- Pour la prise en charge des transactions à configuration groupée, toutes les applications prenant part à la même transaction XA doivent utiliser la même instance de serveur DB2 Connect pour se connecter à l'hôte.
- Seules les applications qui ferment les ressources mises en attente (telles que les curseurs placés en attente) aux termes des transactions peuvent tirer partie du concentrateur. Les transactions qui ne ferment pas les curseurs placés en attente cesseront d'être acceptées et se verront assigner un agent exécutant dédié et ne seront, par conséquent, plus capables d'utiliser le jeu complet des fonctions du concentrateur.
- Si vous déclarez des tables temporaires, elles doivent être supprimées explicitement à la démarcation de la transaction ou de la branche. Si vous l'omettez, la concentration de connexion sera désactivée, bien que l'application continue à fonctionner.

- Toute application prenant part à la même transaction XA doit posséder le même CCSID et utiliser le même ID utilisateur pour établir la connexion.
- Si une connexion sortante a été établie afin de prendre en charge la connexion en deux phases, cet agent de connexion peut uniquement être utilisé pour prendre en charge les connexions en deux phases. De même, les agents établis pour prendre en charge une connexion en une phase peuvent uniquement prendre en charge des connexions en une phase.
- Le concentrateur prend uniquement en charge le SQL dynamique de l'interface CLI (Call Level Interface). Les applications CLI ne doivent également pas utiliser KEEP DYNAMIC car le concentrateur se base sur les instructions en cours de nouvelle préparation au terme de chaque transaction.
- Les requêtes prepare dynamiques issues des applications SQL imbriquées dynamiques seront refusées. Vos applications doivent être modifiées pour utiliser le SQL statique ou l'interface CLI pour les instructions SQL dynamiques.
- Si le concentrateur de connexion est ON (activé), la requête entrante sur le serveur DB2 Connect ne peut pas utiliser SSL. La requête sortante vers le serveur de base de données cible peut toutefois utiliser SSL. Si le concentrateur de connexion est OFF (désactivé), la requête entrante et sortante peuvent toutes deux utiliser SSL.

Lors de l'utilisation de DB2 version 9 ou version 8 avec Groupes de correctifs 13 (ou ultérieur), l'activation de la prise en charge du concentrateur de DB2 Connect requiert IBM Power Systems version 5 édition 4 (PTF SI23726). Sinon, seule la partie XA du concentrateur de connexion est prise en charge.

Activation du concentrateur de connexion

Le paramètre de configuration du gestionnaire de base de données *max_coordagents* définit le nombre maximal d'agents logiques. Vous pouvez activer la fonction du concentrateur de connexion en définissant la valeur de *max_connections* sur un nombre supérieur à la valeur par défaut. La valeur par défaut de *max_connections* est identique à la valeur de *max_coordagents*. Puisque chaque application ne disposera que d'un seul agent logique, *max_connections* contrôle le nombre d'applications pouvant être connectées à l'instance de base de données alors que *max_coordagents* contrôle le nombre de connexions entrantes pouvant être actives en même temps. Le paramètre *max_connections* prendra une valeur numérique comprise entre *max_coordagents* et 64 000. Le nombre par défaut d'agents logiques est égal *max_coordagents*.

max_connections et *max_coordagents* peuvent tous deux être définis sur AUTOMATIC. Si *max_connections* a été défini sur AUTOMATIC, le nombre de connexions peut augmenter au delà de la valeur de base configurée. Si *max_connections* et *max_coordagents* ont tous deux été définis sur AUTOMATIC, *max_connections* pourra croître au delà de sa valeur de base et *max_coordagents* sera augmenté automatiquement afin de conserver le rapport de concentration entre les connexions et les agents de coordination.

Plusieurs paramètres de configuration existants sont utilisés pour configurer les agents. Il s'agit des paramètres suivants :

max_coordagents

Nombre maximal d'agents de coordination actifs.

num_poolagents

Taille du regroupement d'agents. Le regroupement d'agents englobe les agents inactifs et mis en veille. Pour améliorer les performances, *num_poolagents* doit être configuré sur une valeur égale au nombre moyen de clients.

num_initagents

Nombre initial d'agents exécutants dans le regroupement. Il s'agira des agents mis en veille.

Support des transactions XA

L'architecture du concentrateur de connexion permet à DB2 Connect d'assurer un couplage étroit des transactions XA entre DB2 for z/OS et DB2 for IBM i. Le concentrateur associe un agent exécutant à une transaction XA définie (XID unique) comme il le ferait pour n'importe quelle autre transaction. Cependant, si la transaction XA s'achève par `xa_end()` (terme du branchement), l'agent exécutant ne se libérera pas dans un regroupement général. Au lieu de cela, l'agent exécutant restera associé à cette transaction XA définie. Lorsqu'une autre application se joint à la transaction XA, l'agent exécutant sera également associé à cette application.

Tout appel de fin de transaction entraînera le retour de l'agent dans le regroupement. Par exemple, toute instruction `xa_prepare()` en lecture seulement, `xa_rollback()`, `xa_recover()`, `xa_forget()`, `xa_commit()` ou toute erreur XA entraînant l'annulation provoquera le retour de l'agent dans le regroupement normal. `xa_end()` ne s'achève qu'au terme du branchement de la transaction, ce qui n'est pas suffisant pour mettre un terme à son association avec le XID.

Exemple de prise en charge des transactions XA

1. Envisagez un environnement nécessitant au moins 4000 connexions simultanées. Un serveur Web qui utilise les applications CGI ou un système Office avec de nombreux utilisateurs du bureau peuvent tout deux dépasser cette exigence. Dans ce cas, pour que le système soit efficace, DB2 Connect devra fonctionner comme une passerelle indépendante, c'est-à-dire que la base de données et le système DB2 Connect doivent se trouver sur la même machine.

Il se peut que le serveur DB2 Connect ne soit pas en mesure de gérer 4000 connexions ouvertes simultanément dans la machine de base de données. Dans la plupart des cas, le nombre de transactions établies à n'importe quel moment sera considérablement inférieur au nombre de connexions concurrentes.

L'administrateur système peut alors optimiser l'efficacité du système en définissant les paramètres de configuration de la base de données comme suit :

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

Le concentrateur conservera ouvertes les 4000 connexions simultanées, même si la passerelle ne gère que 1000 transactions à la fois.

2. Dans l'exemple susmentionné, les agents exécutants forment et rompent constamment des associations avec les agents logiques. Ces agents qui ne sont pas en veille peuvent gérer une connexion à la base de données mais ne prennent part à aucune transaction particulière ; ils sont par conséquent disponibles pour n'importe quel agent logique (application) qui demande une connexion.

Le cas des transactions est quelque peu différent. Dans cet exemple, supposons qu'un moniteur TP est utilisé avec une passerelle DB2 Connect et une base de données System z ou IBM Power Systems. Lorsqu'une application demande une connexion, le concentrateur active un agent inactif pour prendre en charge la requête ou crée un nouvel agent exécutant. Supposons que l'application demande une transaction XA. Un XID (ID transaction) est créé pour cette transaction et l'agent exécutant est associé à cet XID.

Lorsque la demande de l'application a été traitée, elle émet un `xa_end()` et se désassocie de l'agent exécutant. L'agent exécutant reste associé au XID de la transaction. Il ne peut désormais prendre en charge que des demandes de transaction possédant cet XID associé.

Une autre application peut alors demander une transaction non XA. Même si aucun autre agent exécutant n'est disponible, l'agent associé au XID ne sera pas disponible pour cette seconde application. Il est considéré comme actif. Un nouvel agent exécutant sera créé pour cette seconde application. Lorsque cette application termine sa transaction, son agent d'exécution est libéré dans le regroupement disponible.

Entre-temps, d'autres applications requérant la transaction associée au premier XID de l'agent peuvent s'associer et se désassocier de l'agent qui exécute sa transaction XA dédiée pour elles. Toute application requérant cette transaction particulière sera envoyée à cet agent exécutant s'il est libre.

L'agent exécutant ne sera pas libéré dans un regroupement général tant que l'application n'émettra pas un appel de fin de transaction (autre que `xa_end()`). Par exemple, une application peut mettre un terme à la transaction à l'aide de `xa_commit()`, moment à partir duquel l'agent exécutant rompt son association avec le XID et retourne dans le regroupement disponible. A ce moment, toute application demandeuse peut l'utiliser pour traiter une transaction XA ou non XA.

Regroupement et concentrateur de connexions

Alors que le regroupement de connexions et le concentrateur de connexion semblent posséder des similitudes, ils diffèrent dans leur mise en oeuvre et traitent des problèmes différents. Le regroupement de connexions aide à réduire le temps système des connexions à la base de données et à gérer le volume des connexions. Le concentrateur de connexion facilite l'évolutivité de votre solution DB2 for z/OS et DB2 Connect en optimisant l'utilisation de vos serveurs de base de données hôte.

Lorsque vous utilisez le regroupement de connexions, la connexion est toujours disponible afin d'être réutilisée une fois que l'application qui possède la connexion a émis une demande de déconnexion. Dans de nombreuses applications client-serveur à deux niveaux, les utilisateurs ne se déconnectent pas de toute la journée. Aussi, la plupart des serveurs d'applications dans des applications multiniveau établissent des connexions à la base de données au démarrage du serveur et ne libèrent pas ces connexions avant la fermeture du serveur d'applications.

Dans ces environnements, le regroupement de connexions n'apportera que peu, voire aucune amélioration. Cependant, dans des environnements client-serveur dans lesquels la fréquence de connexions et de déconnexions est plus élevée, le regroupement de connexions apportera une amélioration considérable des performances. Le concentrateur de connexion attribue des ressources de base de données hôte uniquement pendant la durée de la transaction SQL tout en conservant les applications utilisateur actives. Cela permet de définir des configurations dans lesquelles le nombre d'unités d'exécution et de ressources DB2 consommés peut être considérablement réduit par rapport à des configurations dans lesquelles chaque connexion d'application possède sa propre unité d'exécution.

Lorsqu'il s'agit d'opérations insensibles aux pannes et de la répartition des charges de travail, le concentrateur de connexion est réellement la meilleure option car il permet de réaffecter du travail lors de chaque nouvelle transaction. De même, le regroupement de connexions ne peut offrir qu'une répartition des charges très limitée et ce uniquement à la connexion.

Le regroupement de connexions et le concentrateur de connexion doivent être utilisés conjointement bien qu'ils abordent chacun des problèmes différents.

Un concentrateur de connexion est requis avec WebSphere MQ Transaction Manager et DB2 for z/OS

Lors de l'exécution d'applications dans un environnement IBM WebSphere MQ (dénommé auparavant IBM MQSeries), WebSphere MQ peut opérer en tant que gestionnaire de transactions compatible XA, en coordonnant toutes les transactions de validation réparties, à deux phases. Lorsque WebSphere MQ remplit ce rôle et que les sources de données proviennent de la famille de produits DB2, diverses exigences de configuration sont requises.

La majeure partie de la configuration requise pour un environnement de gestionnaire de transactions est décrite ailleurs dans la documentation. Vous devez, par exemple, définir le paramètre de configuration DB2 TP_MON_NAME à «MQ» sur le client d'exécution DB2.

Une exigence de configuration n'était documentée nulle part jusqu'à présent. Elle est spécifique à DB2 Connect en cas de connexion à des sources de données qui sont des serveurs DB2 for z/OS : lors de l'utilisation de WebSphere MQ pour coordonner des transactions réparties impliquant des serveurs DB2 for z/OS et DB2 for IBM i, la fonction de concentrateur de connexion de DB2 Connect doit être activée au niveau de la passerelle. L'activation du concentrateur de connexion s'effectue quand la valeur du paramètre de configuration **max_connections** est supérieure à celle du paramètre **max_coordagents**.

Si vous n'activez pas le concentrateur de connexion, un comportement de transaction inattendu se produit.

Chapitre 18. Prise en charge de Sysplex par le serveur DB2 Connect

Un Sysplex est un ensemble de serveurs System z qui coopèrent en utilisant des matériels et des logiciels pour exécuter des travaux. Il coordonne la coopération en augmentant le nombre de processeurs fonctionnant conjointement, et donc le volume de travail pouvant être traité. En plus d'une capacité de traitement accrue, le Sysplex permet d'utiliser avec souplesse du matériel et des logiciels de différents niveaux et d'ajouter des systèmes de façon dynamique.

Le Sysplex permet à DB2 Connect d'assurer un équilibrage transparent des connexions entre les différents membres d'un groupe partageant des données. Par ailleurs, grâce au Sysplex, DB2 Connect dispose des moyens nécessaires pour tenter un passage de relais entre membres au cas où l'un des membres vient à rencontrer un problème. La fonction de redirection pour Sysplex est une option DB2 Connect. Le support DB2 Connect pour Sysplex est activé par défaut, de même que la fonction de redirection pour Sysplex. Le support Sysplex d'une base de données hôte peut être désactivé en supprimant le paramètre SYSPLEX de son répertoire DCS mais l'entrée DCS elle-même ne doit pas être supprimée, même lorsqu'aucun autre paramètre n'est défini.

Avec la fonction de redirection client automatique pour Sysplex, le comportement par défaut pour une connexion Sysplex activée consiste à tenter une nouvelle connexion en cas d'incident de communication. Les valeurs de registre spéciales, jusqu'à la dernière transaction réussie n'immobilisant pas de ressources, sont relues lorsque DB2 Connect est connecté à un serveur DB2 for z/OS.

A l'aide des variables de registre DB2_MAX_CLIENT_CONNRETRIES et DB2_CONNRETRIES_INTERVAL, il vous est possible de définir le comportement exactement souhaité pour les nouvelles tentatives de redirection client automatique, y compris la désactivation. La variable de registre pour le délai de connexion est DB2TCP_CLIENT_CONTIMEOUT.

Considérations concernant l'exploitation de SYSPLEX sur System z

DB2 Connect assure la répartition des charges et la tolérance aux pannes lorsqu'il achemine des connexions vers des Sysplex multiples. Lorsque vous êtes connecté à un serveur de base de données DB2 for z/OS opérant dans un environnement de partage des données, DB2 Connect répartit la charge de travail entre les différents sous-systèmes DB2 constituant le groupe de partage en fonction des informations sur la charge de travail fournies par WLM (Workload Manager).

DB2 Connect reçoit de WLM une liste priorisée des membres Sysplex. Chaque Sysplex renvoie des informations relatives à la priorité pondérée pour chaque adresse de connexion. Le serveur DB2 Connect utilise ensuite cette liste pour traiter les requêtes CONNECT entrantes en les distribuant entre les membres Sysplex avec les niveaux de priorité les plus élevés. La liste des informations relatives à la priorité pondérée des Sysplex est extraite à chaque connexion pour assurer la répartition de la charge. Si le concentrateur de connexion DB2 Connect est activé, cette liste sert également à déterminer la destination de chaque transaction.

Remarque : Il n'est pas nécessaire de modifier la configuration System z DDF (Distributed Data Facility) pour tirer partie de DB2 Connect Sysplex.

DB2 Connect assure aussi la tolérance aux pannes en tentant de se connecter à une autre machine Sysplex en cas d'échec de la connexion. Une erreur ne sera renvoyée à l'application que si toutes les tentatives de connexion ont échoué.

Le sysplex DB2 Connect a été conçu en avec le concept de regroupement d'agents à l'esprit. Lorsque Sysplex est activé, DB2 Connect dirige les connexions vers un autre membre DDF si la connexion vers un membre participant est perdue. La redirection est effectuée conformément à une liste de serveurs Sysplex.

Moyennant l'ajout d'un concentrateur, DB2 Connect est désormais capable d'équilibrer la charge au niveau des frontières de la transaction. Le concentrateur DB2 Connect doit être activé pour ce faire.

Exploitation de Sysplex avec DB2

Pour reprendre un exemple classique, un serveur DB2 Connect (serveur A) converse avec un Sysplex contenant deux serveurs DB2 for z/OS (les serveurs B et C).

Serveur Sysplex B	Serveur Sysplex C
HOST_NAME=MVSHOST	HOST_NAME=MVSHOST1

Supposons maintenant qu'une application lance la commande :

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

La connexion à la base de données MVSHOST est établie. L'exploitation Sysplex étant activée tant pour le serveur DB2 Connect que pour l'entrée d'annuaire DCS, DB2 for z/OS identifie les adresses réseau auprès de DB2 Connect pour chaque participant Sysplex (MVSHOST et MVSHOST1). Les protocoles DRDA4 et les flux de message sont utilisés pour renvoyer cette information. Une fois la connexion initiale établie, la liste des adresses renvoyées est placée dans la mémoire cache du poste de travail DB2 Connect. Par exemple, si la commande CONNECT est lancée pour un noeud APPPC TCP/IP, seules les adresses IP sont renvoyées.

Informations sur la priorité utilisées pour la répartition de la charge et la tolérance aux pannes

La liste d'adresses fournie par DB2 for z/OS comprend également des informations de priorité, notamment le nombre de connexions pour chaque adresse réseau. Cette liste est régénérée chaque fois que DB2 Connect établit une nouvelle connexion. Ces informations supplémentaires sont utilisées pour l'équilibrage de charge et la tolérance aux pannes.

Liste d'adresses en cache utilisée par DB2 Connect

Si la connexion de la base de données à ALIASB échoue, un message d'erreur SQL30081N s'affiche et la connexion est interrompue. Si une nouvelle demande de connexion à ALIASB est reçue, DB2 Connect effectue les opérations suivantes :

1. Il essaie le serveur disposant de la priorité la plus élevée dans la liste d'adresses en cache en fonction des informations de priorité renvoyées par DB2 for z/OS. Cette stratégie est toujours celle de DB2 Connect, et c'est par ce moyen que l'équilibrage de charge est effectué.
2. Si cette tentative de connexion échoue, d'autres adresses sont essayées successivement, par ordre de priorité décroissante, telles que renvoyées par DB2 for z/OS. C'est ainsi que DB2 Connect tire parti des informations Sysplex pour obtenir la tolérance aux pannes.
3. Lorsque toutes les autres tentatives de connexion ont échoué, DB2 Connect tente à nouveau de se connecter à ALIASB au moyen de l'adresse contenue dans le répertoire des noeuds catalogué.

La commande db2pd avec le paramètre `sysplex` (`db2pd -sysplex`) peut être utilisée pour extraire des informations relatives aux serveurs associés à un environnement Sysplex.

Configuration requise pour Sysplex

L'exploitation du support Sysplex ne sera utilisée avec une base de données que si l'entrée du répertoire DCS pour cette base est Sysplex (pas de distinction majuscules/minuscules) dans le sixième paramètre positionnel.

Chapitre 19. Support d'un Sysplex client

Les clients de serveur de données IBM et les pilotes de serveur de données non Java disposant d'une licence DB2 Connect peuvent accéder à un Sysplex DB2 for z/OS et exploiter les capacités Sysplex complètes, notamment l'équilibrage de charge au niveau des transactions, la redirection automatique du client et une prise en charge XA restreinte.

Un Sysplex est un ensemble de systèmes DB2 (connus également sous le nom de membres) qui constituent un groupe de partage de données. Une ou plusieurs unités de couplage offrent des fonctions de mise en cache à grande vitesse et de traitement des verrous pour le groupe de partage de données. Le Sysplex ainsi que le gestionnaire de charge de travail WLM (Workload Manager), l'adresse IP virtuelle dynamique (DVIPA) et Sysplex Distributor permettent à un client d'accéder à une base de données DB2 for z/OS sous TCP/IP avec fonction de résilience des réseaux et distribuent les transactions d'une application en respectant un certain équilibre, à tous les membres du groupe de partage de données.

L'essentiel de ces fonctionnalités repose sur une liste de serveurs que chaque membre du groupe de partage de données DB2 renvoie concernant les limites de connexion et les limites de transaction (facultatif). Cette liste contient l'adresse IP et la capacité disponible pour chaque membre DB2. Grâce à ces informations, un client peut distribuer des transactions en respectant un certain équilibre, ou identifier le membre DB2 à utiliser en cas d'incident de communication.

La liste des serveurs est renvoyée vers la base de données DB2 une fois la première connexion établie. Par conséquent, la connexion de base de données initiale devrait être acheminée vers l'adresse IP virtuelle dynamique (DVIPA) du groupe appartenant à Sysplex Distributor. Si au moins un membre DB2 est disponible, Sysplex Distributor acheminera la requête vers la base de données. Après que le client a reçu la liste des serveurs, celui-ci accède directement à un membre DB2 en tenant compte des informations contenues dans la liste des serveurs.

Pour configurer le support d'un Sysplex client, vous devez définir des paramètres dans le fichier de configuration db2dsdriver.

Pour plus d'informations sur WLM, DVIPA et Sysplex Distributor, reportez-vous au centre de documentation DB2 for z/OS à l'adresse <http://publib.boulder.ibm.com/infocenter/dzichelp>.

Équilibrage de charge au niveau des transactions (côté client)

Les clients de serveur de données IBM et de pilotes de serveur de données non Java qui détiennent une licence DB2 Connect peuvent exécuter des tâches d'équilibrage de charge de travail au niveau des transactions sans devoir passer par le serveur DB2 Connect. Grâce à cette fonctionnalité, la charge de travail d'un membre d'un Sysplex DB2 for z/OS peut être rééquilibrée au démarrage d'une transaction.

L'équilibrage de la charge du travail est exécutée sur des transactions simultanées sur une connexion unique ou sur plusieurs connexions par application. Pour une application qui invoque uniquement une connexion à exécution longue sur laquelle des transactions s'exécutent les unes après les autres plutôt qu'en simultanée, la charge de travail est reportée sur le membre dont la capacité est la plus élevée. L'algorithme de l'équilibrage de la charge de travail s'ajuste automatiquement aux changements se produisant au niveau du volume de travail.

La présentation suivante décrit les étapes se produisant lorsqu'un client se connecte à un Sysplex DB2 for z/OS, et que l'équilibrage de la charge de travail avec réutilisation du transport entre en jeu :

1. Lorsque le client établit une connexion, le serveur génère une liste de serveurs contenant des informations relatives à la distribution de la charge de travail d'un membre.
La liste des serveurs est mise en cache par le client et dispose d'une durée de vie configurable définie par défaut à 30 secondes.
2. Au démarrage d'une nouvelle transaction, le client consulte la liste des serveurs mis en cache afin d'identifier un membre avec une capacité inexploitée et recherche dans leur pool un transport en veille lié au membre avec la capacité sous-utilisée. (Un transport en veille est un transport ne disposant d'aucune connexion logique associée)
 - Si un transport en veille est disponible, le client associe la connexion logique au transport.
 - Si, après un dépassement du délai d'attente configurable par l'utilisateur, aucun transport en veille n'est disponible dans le pool de transport et qu'aucun nouveau transport ne peut être attribué car le pool de transport a atteint sa limite, une erreur est renvoyée à l'application.
3. Lors de l'exécution de la transaction, elle accède au membre lié au transport.
4. Lorsque la transaction aboutit, le client vérifie avec le serveur que la réutilisation du transport est toujours autorisée pour la connexion logique.
5. Si tel est le cas, le serveur génère une liste des instructions SET pour les registres spéciaux qui s'appliquent à l'environnement d'exécution pour la connexion logique.
Le client met ces instructions en cache, qu'il exécute de nouveau afin de reconstruire l'environnement d'exécution lorsque la connexion logique est associée à un nouveau transport.
6. Si la réutilisation du transport est autorisée, la connexion logique est périodiquement dissociée du transport et la copie client de la liste des serveurs est actualisée.
7. Lorsque l'équilibrage de charge de travail est requis pour une nouvelle transaction, le client utilise le même processus pour associer la connexion logique à un transport.

Conseil : L'équilibrage de charge au niveau des transactions côté client est effectué sur les transactions s'exécutant dans un processus d'application unique. Si plusieurs processus applicatifs s'exécutent sur le client accèdent à la même base de données activée sur le Sysplex, l'équilibrage de charge est exécuté pour chaque processus applicatif indépendamment des autres. Pour effectuer un équilibrage de charge au niveau transactions sur plusieurs connexions émanant d'applications ou de processus multiples s'exécutant sur plusieurs clients distants, utilisez les fonctions d'équilibrage de charge du travail disponibles dans le serveur DB2 Connect.

Configuration de l'équilibrage de charge au niveau des transactions (côté client)

Pour configurer l'équilibrage de charge des transactions côté client, définissez les paramètres appropriés dans le fichier de configuration db2dsdriver.

Avant de commencer

Pour exécuter l'équilibrage de charge au niveau des transactions, le client doit utiliser une connexion TCP/IP et disposer d'une licence DB2 Connect. Les clients suivants prennent en charge l'équilibrage de charge au niveau des transactions :

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

A propos de cette tâche

Cette tâche configure l'équilibrage de charge au niveau des transactions, ce qui permet au client d'exécuter cet équilibrage sans avoir à passer par le serveur DB2 Connect server.

Procédure

1. Dans le fichier de configuration db2dsdriver, activez l'équilibrage de charge au niveau des transactions en définissant le paramètre enableWLB à "true" dans la sous-section WLB d'une entrée de base de données ou d'une entrée DSN. Par exemple, spécifiez ce qui suit dans le fichier de configuration db2dsdriver

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">  
  <!-- database-specific parameters -->  
  <WLB>  
    <!-- Sysplex WLB is disabled by default -->  
    <parameter name="enableWLB" value="true" />  
  </WLB>  
</database>
```

Par défaut, enableWLB est défini à 'false' et l'équilibrage de charge est désactivé.

2. Facultatif : Affinez les paramètres d'équilibrage de charge en spécifiant des valeurs appropriées pour les paramètres suivants. Les valeurs par défaut de ces paramètres devraient convenir à la plupart des applications.

Tableau 17. Paramètres d'équilibrage de charge de travail dans le fichier de configuration db2dsdriver

Paramètre	Description
maxTransports	Indique le nombre maximal de transports dans un pool de transports. Par défaut, la valeur est -1 (aucune limite). Toute autre valeur négative n'est pas valide. La valeur 0 désactivera l'équilibrage de charge de travail.
maxTransportIdleTime	Indique le temps maximal (exprimé en secondes) écoulé avant qu'un transport mis en veille ne soit supprimé. La valeur par défaut est 600. La valeur minimale prise en charge est 0.

Tableau 17. Paramètres d'équilibrage de charge de travail dans le fichier de configuration db2dsdriver (suite)

Paramètre	Description
maxTransportWaitTime	Indique le délai en secondes devant s'écouler avant que le client puisse utiliser un nouveau transport. Par défaut, la valeur est -1 (aucune limite). La valeur minimale prise en charge est 0.
maxRefreshInterval	Indique la durée maximale écoulée exprimée en secondes avant que la liste des serveurs ne soit actualisée. La valeur par défaut est 30. La valeur minimale prise en charge est 0.

Pour plus d'informations, reportez-vous à la rubrique dédiée au fichier de configuration db2dsdriver dans le centre de documentation DB2.

Remarque : Les paramètres de configuration du gestionnaire de base de données MAX_COORDAGENTS, NUM_POOLAGENTS et MAX_CONNECTIONS servent à configurer l'équilibrage de charge de travail uniquement au niveau du serveur DB2 Connect et n'affecte pas l'équilibrage de charge de travail effectué côté client.

Exemple

Exemple 1 : Activation de l'équilibrage de charge au niveau des transactions pour la base de données SAMPLE en utilisant les paramètres par défaut.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- database-specific parameters -->
  <WLB>
    <!-- Sysplex WLB is disabled by default -->
    <parameter name="enableWLB" value="true" />
  </WLB>
</database>
```

Exemple 2 : Activation de l'équilibrage de charge au niveau des transactions pour la base de données SAMPLE avec un maximum de 95 transports.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- database-specific parameters -->
  <WLB>
    <!-- Sysplex WLB is disabled by default -->
    <parameter name="enableWLB" value="true" />
    <!-- maxTransports represents the maximum number of transports;
    defaults to -1 (unlimited) -->
    <parameter name="maxTransports" value="95" />
  </WLB>
</database>
```

Exemple 3 : Activation de l'équilibrage de charge au niveau des transactions pour la base de données SAMPLE avec un maximum de 80 transports et un délai d'attente maximal de 10 secondes pour un transport disponible.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- database-specific parameters -->
  <WLB>
    <!-- Sysplex WLB is disabled by default -->
    <parameter name="enableWLB" value="true" />
    <!-- maxTransports represents the maximum number of transports;
    defaults to -1 (unlimited) -->
    <parameter name="maxTransports" value="80" />
    <!-- maxTransportWaitTime represents the maximum elapsed time in
```

```

        number of seconds before a transport becomes available;
        defaults to -1 (unlimited)-->
        <parameter name="maxTransportWaitTime" value="10" />
    </WLB>
</database>

```

Exemple 4 : Désactivation de l'équilibrage de charge au niveau des transactions pour la base de données SAMPLE.

```

<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
    <!-- database-specific parameters -->
    <WLB>
        <!-- Sysplex WLB is already disabled by default -->
        <parameter name="enableWLB" value="false" />
    </WLB>
</database>

```

Redirection automatique du client (côté client)

La fonction de redirection automatique du client fournit une fonction de secours lorsqu'un client de serveur de données IBM ne parvient plus à se connecter à un membre d'un Sysplex de DB2 for z/OS. La redirection automatique du client permet au client d'effectuer une opération de reprise sur incident en tentant de se reconnecter à la base de données via l'un des membres disponibles, quels qu'ils soient, du Sysplex.

La fonction de redirection automatique du client est activée par défaut lorsque l'équilibrage de charge est activé.

La prise en charge de la redirection automatique côté client est disponible pour les clients de serveurs de données IBM et pour les pilotes de serveurs de données non Java disposant d'une licence DB2 Connect. Le serveur DB2 Connect n'est pas requis pour la redirection automatique du client.

Si la fonction de redirection automatique du client est activée, le processus suivant se produit normalement lorsqu'un client détecte un incident de connexion avec une connexion existante :

1. Le client tente d'exécuter une instruction SQL à l'aide d'une connexion existante et rencontre un échec.
2. Le client utilise la liste des serveurs pour identifier le Sysplex auquel accéder et tente de se reconnecter à la base de données. Il peut s'agir de la liste renvoyée par le serveur, ou dans le cas où des affinités client ont été spécifiées, d'une liste alternative de serveurs indiquée par le client. Pour plus d'informations, reportez-vous «Configuration des affinités client», à la page 133.
3. Si la fonction de redirection automatique parvient à se connecter à la base de données, le client reconstruit l'environnement d'exécution pour la connexion qui vient d'être établie. L'erreur SQL30108N est renvoyée vers l'application pour indiquer que la récupération de la connexion de la base de données sur laquelle un incident s'était produit a abouti et que la transaction a été annulée. L'application est alors responsable des reprises ultérieures, notamment de la resoumission de toute tâche qui a été annulée.
4. Si la fonction de redirection automatique du client n'est pas en mesure de se reconnecter à la base de données, l'erreur SQL30081N est renvoyée vers l'application. L'application va alors tenter d'effectuer une opération de récupération suite à l'incident de connexion (par exemple, en tentant de se reconnecter par elle-même à la base de données).

La fonction de redirection automatique du client est également utilisée lorsqu'un client détecte un incident de connexion survenu sur une nouvelle connexion. Toutefois, si tel est le cas, l'erreur SQL30108N n'est pas renvoyée vers l'application pour indiquer que l'échec de la connexion de la base de données a été résolu. Soit la connexion a abouti, soit l'erreur SQL30081N est renvoyée.

Vous pouvez désactiver la fonction de redirection automatique ou affiner sa configuration en spécifiant des paramètres dans le fichier de configuration db2dsdriver.

Reprise en ligne en continu pour les applications CLI et .NET

Lorsque la fonction de redirection automatique du client est activée et que la cible de la transaction est DB2 for z/OS, la fonction de reprise en ligne en continu pour les applications CLI et .NET est activée par défaut. Avec la fonction de reprise en ligne en continu, si une application rencontre un échec de la connexion lors de la première opération SQL dans une transaction, le pilote relance l'opération SQL ayant échoué dans le cadre de l'exécution de la fonction de redirection automatique du client. Si la connexion aboutit, aucune erreur n'est renvoyée vers l'application et la transaction n'est pas annulée. L'incident de connectivité et la reprise en découlant sont masqués.

Vous pouvez désactiver ou réactiver la fonction de reprise en ligne en continu en spécifiant certains paramètres dans le fichier de configuration db2dsdriver.

Considérations du serveur DB2 Connect pour la fonction de redirection automatique du client et la fonction de reprise en ligne en continu

Le comportement de la fonction de redirection automatique du client et la fonction de reprise en ligne en continu est différent si le client accède ou non au Sysplex via un serveur DB2 Connect.

Si l'incident se produit entre le serveur DB2 Connect et le Sysplex, la fonction de redirection automatique du client est exécutée par le serveur DB2 Connect. Si le niveau de version du serveur DB2 Connect est le même que le niveau de version du client, ou un niveau supérieur, le client peut exécuter la fonction de reprise en ligne en continu. Dans le cas contraire, le client n'exécute pas la fonction de reprise en ligne en continu et l'erreur SQL30108N est renvoyée vers l'application pour indiquer que la récupération de la connexion de la base de données a abouti.

Si l'incident se produit entre le client et le serveur DB2 Connect, la fonction de redirection automatique du client peut être exécutée côté client vers le serveur DB2 Connect. Cependant, la fonction de reprise en ligne en continu est toujours désactivée et l'erreur SQL30108N est renvoyée à l'application.

Configuration de la redirection automatique du client (côté client)

La fonction de redirection automatique du client est activée par défaut lorsque l'équilibrage de charge est activé. Vous pouvez la désactiver ou affiner sa configuration en spécifiant des paramètres dans le fichier de configuration db2dsdriver.

Avant de commencer

Pour exécuter la redirection automatique du client, le client doit utiliser une connexion TCP/IP et être détenteur d'une licence DB2 Connect. Les clients suivants prennent en charge la fonction de redirection automatique du client :

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

A propos de cette tâche

Dans le cas de certaines applications, vous voudrez peut être désactiver la fonction de redirection automatique du client ou la reprise en ligne en continu, ou affiner la configuration de la redirection automatique. Cette tâche décrit les paramètres disponibles pour configurer la fonction de redirection automatique du client.

Vous pouvez également modifier le comportement de cette fonction en spécifiant des affinités client pour contrôler la liste des serveurs utilisés pour le traitement des transactions et de la redirection automatique du client. Pour plus d'informations, voir «Configuration des affinités client», à la page 133.

Procédure

Pour désactiver la redirection automatique du client ou affiner sa configuration, procédez comme suit :

- Pour désactiver la redirection automatique du client ou la redirection automatique du client avec reprise en continu, spécifiez la valeur "false" pour les paramètres suivantes dans le fichier de configuration db2dsdriver. Vous indiquez ces paramètres dans la sous-section relative à la redirection automatique du client dans une entrée de base de données.

Tableau 18. Paramètres de redirection automatique du client dans le fichier de configuration db2dsdriver

Paramètre	Description
enableAcr	Indique si la fonction de redirection automatique du client est activée (true) ou désactivée (false). Ce paramètre reçoit par défaut la valeur 'true' lorsque l'équilibrage de charge au niveau des transactions est activé (lorsque le paramètre enableWLB est défini à 'true'). Si la fonction de redirection automatique est désactivée, tous les autres paramètres sont ignorés.
enableSeamlessAcr	Indique si la fonction de redirection automatique du client dotée de la fonction de reprise en ligne en continu est activée (true) ou désactivée (false). Si elle est activée et que le serveur cible est DB2 for z/OS, la fonction de redirection automatique du client est activée par défaut. Si le serveur cible est différent de DB2 for z/OS, ce paramètre est ignoré et la fonction de redirection automatique du client n'est pas prise en charge.

Pour plus d'informations, reportez-vous à la rubrique dédiée au fichier de configuration db2dsdriver dans le centre de documentation DB2.

- Vous pouvez spécifier une liste de serveurs alternatifs à utiliser lorsqu'un échec survient à la première tentative de connexion et que l'hôte spécifié dans la connexion n'est pas disponible. Pour spécifier une liste de serveurs alternatifs, attribuez la valeur "true" à enableAlternateServerListFirstConnect dans la sous-section de redirection automatique du client d'une entrée de base de données. Lorsque enableAlternateServerListFirstConnect est défini à "true," la redirection automatique du client avec reprise en ligne en continu est activée implicitement quels que soient les autres paramètres spécifiés pour cette fonction dans la fichier de configuration db2dsdriver. Pour utiliser cette fonction, vous devez également stipuler une liste de serveurs alternatifs dans le fichier de configuration db2dsdriver.

Pour plus d'informations sur la définition d'une liste de serveurs alternatifs, voir «Configuration des affinités client», à la page 133. Bien que les informations sur la marche à suivre figurent dans la rubrique décrivant les affinités client, vous n'avez pas besoin d'utiliser des affinités client pour exploiter cette capacité. En fait, cette capacité est également prise en charge lorsque l'équilibrage de charge au niveau des transactions est activé, bien que cet équilibrage et les affinités client ne puissent pas être activés simultanément.

Résultats

Si la fonction de redirection automatique du client est activée, une application sur laquelle survient un incident de connectivité est réacheminée vers un autre membre du Sysplex. Si la reprise en ligne en continu est désactivée ou non disponible, l'erreur SQL30108N est renvoyée à l'application.

Si l'incident se produit sur la première opération SQL d'une transaction et que la reprise en ligne en continu est activée (et disponible), le client peut exécuter à nouveau l'opération SQL qui a échoué lors de la redirection automatique du client. Si la connexion aboutit, aucune erreur n'est renvoyée vers l'application.

Exemple

Exemple 1 : Activation de la redirection automatique du client pour la base de données SAMPLE avec reprise en ligne en continu désactivée

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- database-specific parameters -->
  <WLB>
    <!-- Sysplex WLB is disabled by default -->
    <parameter name="enableWLB" value="true" />
  </WLB>
  <ACR>
    <!-- ACR is already enabled by default when WLB is enabled -->
    <parameter name="enableAcr" value="true" />
    <!-- If ACR is enabled and the target server is DB2 for z/OS,
    Seamless ACR is enabled by default; if the target server is
    not DB2 for z/OS, this setting is ignored and seamless ACR
    is not supported. -->
    <parameter name="enableSeamlessAcr" value="false" />
  </ACR>
</database>
```

Exemple 2 : Désactivation de la redirection automatique du client pour la base de données SAMPLE

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- database-specific parameters -->
  <WLB>
    <!-- Sysplex WLB is disabled by default -->
    <parameter name="enableWLB" value="true" />
  </WLB>
  <ACR>
    <!-- ACR is enabled by default when WLB is enabled -->
    <parameter name="enableAcr" value="false" />
  </ACR>
</database>
```

Exemple 3 : Directive d'utilisation de la liste de serveurs alternatifs lors de la première connexion si l'hôte original n'est pas disponible

```
<database name="SAMPLE" host="v33ec065.svl.ibm.com" port="446">
  <ACR>
    <parameter name="enableACR" value="true"/>
    <parameter name="enableAlternateServerListFirstConnect"
      value="true"/>
  </ACR>
</database>
```

Support XA (côté client)

Les clients de serveurs de données IBM et les pilotes de serveurs de données non Java disposant d'une licence DB2 Connect peuvent accéder directement à un Sysplex DB2 for z/OS et utiliser le support Sysplex natif sans devoir passer par un serveur DB2 Connect de niveau intermédiaire.

Ce type de support XA côté client n'est disponible que pour les gestionnaires de transactions utilisant un modèle de traitement à transport unique. Dans un modèle à transport unique, une transaction acheminée via un transport unique (connexion physique) est liée à un membre depuis `xa_start` jusqu'à `xa_end`. La fin de la transaction est immédiatement suivie de `xa_prepare` (en lecture seule), `xa_prepare` plus `xa_commit` ou `xa_rollback`, ou `xa_rollback`. Tout ceci doit se produire dans le cadre d'un processus applicatif unique. Parmi les gestionnaires de transaction utilisant ce modèle, on trouve IBM TXSeries CICS, IBM WebSphere Application Server et Microsoft Distributed Transaction Coordinator.

Le support pour le modèle de traitement à transport unique inclut également la récupération de transactions en attente de validation où les informations du membre pour chaque transaction pouvant être récupérée sont extraites via l'opération `xa_recover`, qui permet à l'opération `xa_commit` ou `xa_rollback` d'être acheminée vers le membre spécifié.

Vous pouvez activer le support XA en exécutant le paramètre `SINGLE_PROCESS` dans la chaîne `xa_open` string ou en définissant les paramètres pour XA dans le fichier de configuration `db2dsdriver`.

Important : Le correctif DB2 for z/OS APAR PK69659 doit être installé pour le support XA direct (requis pour les gestionnaires de transaction tels que Microsoft Distributed Transaction Coordinator). Pour plus d'informations, voir APAR PK69659.

Activation du support XA (côté client)

Pour activer le support XA de sorte que les clients puissent avoir accès à un Sysplex DB2 for z/OS, indiquez les paramètres dans le fichier de configuration db2dsdriver ou utilisez le paramètre SINGLE_PROCESS dans la chaîne xa_open.

Avant de commencer

Une licence DB2 Connect est requise pour accéder à un Sysplex DB2 for z/OS.

Les clients suivants assurent une prise en charge XA pour les applications accédant à un Sysplex DB2 for z/OS :

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

Important : Le correctif DB2 for z/OS APAR PK69659 doit être installé pour le support XA direct (requis pour les gestionnaires de transaction tels que Microsoft Distributed Transaction Coordinator). Pour plus d'informations, voir APAR PK69659.

A propos de cette tâche

Cette tâche active le support XA pour les clients de serveur de données IBM et les pilotes de serveur de données non Java.

Restrictions

Le support XA n'est disponible que pour les gestionnaires de transactions utilisant un modèle de traitement à transport unique. Pour plus d'informations sur cette restriction, consultez la rubrique concernant les limitations s'appliquant à un Sysplex client.

Procédure

1. Pour des clients basés sur une instance (clients de serveur de données IBM), indiquez si le support XA est défini sur activé (true) ou désactivé (false) en configurant le paramètre enableDirectXA du fichier de configuration db2dsdriver, ou en utilisant le paramètre SINGLE_PROCESS dans la chaîne xa_open.
2. Pour les clients dépourvus d'une instance, (pilotes de serveur de données IBM), le support XA est activé par défaut pour Microsoft Distributed Transaction Coordinator ou les composants de services Microsoft (COM+). Pour tous les autres gestionnaires de transactions pris en charge, indiquez s'il faut que le support XA soit activé en définissant le mot clé SINGLE_PROCESS dans la chaîne xa_open. Les paramètres d'enableDirectXA dans le fichier de configuration db2dsdriver ne s'appliquent pas aux clients dépourvus d'une instance.

Résultats

Si le support XA est activé, une application peut exécuter une transaction distribuée via un transport unique au sein d'un processus applicatif unique sans devoir passer par un serveur DB2 Connect intermédiaire.

Exemple

Activer le support XA à transport unique pour la base de données SAMPLE.

```
<database name="SAMPLE" host="v33ec065.my.domain.com" port="446">
  <!-- database-specific parameters -->
  <!--directXA is disabled by default -->
  <parameter name="enableDirectXA" value="true" />
</parameters>
</database>
```

Configuration des affinités client

Vous pouvez configurer des affinités client pour déterminer l'ordre des tentatives de contact des serveurs lors du traitement de la connexion initiale et de la redirection automatique du client. Vous pouvez utiliser des affinités client lorsque l'équilibrage de charge au niveau des transactions n'est pas explicitement activé.

A propos de cette tâche

Généralement, les applications qui utilisent les capacités Sysplex déterminent à quel membre se connecter en cas d'échec à l'aide de la liste de serveurs la plus récente renvoyée par le serveur de base de données. Sous ce modèle, le client effectue une redirection d'après la dernière liste de serveurs qu'il a lue et l'ordre de ces serveurs est déterminé par conséquent par le serveur. Vous pouvez cependant supplanter ce comportement en configurant des affinités client et en spécifiant une liste de serveurs alternatifs à utiliser lors du traitement de la connexion initiale et lors du traitement de réacheminement de clients. Lorsque le traitement d'affinités client est actif, l'ordre des tentatives de contact des serveurs pour le traitement de redirection automatique du client est déterminé par le client.

Pour configurer des affinités client, vous devez spécifier des valeurs pour les paramètres `ALTERNATE_SERVER_LIST`, `AFFINITY_LIST`, et `CLIENT_AFFINITY` dans le fichier de configuration `db2dsdriver`. Lorsque ces paramètres sont spécifiés :

- La connexion est établie sur la base de la liste de serveurs alternatifs et non pas avec l'hôte et le port spécifiés par l'application. Par exemple, si l'application spécifie `Host = s1` alors que la liste de serveurs alternatifs spécifie `Serverlist = (s2, s1, s3)`, la connexion initiale est tentée avec `s2` et non pas avec `s1`. L'ordre des tentatives de connexion est le suivant : `(s2,s2,s2) (s1,s1,s1) (s3,s3,s3)`. Le paramètre `acrRetryInterval` est appliqué entre les tentatives de connexion individuelles (pour un total de 8 fois dans cet exemple).
- La redirection transparente du client est automatiquement activée. Cette redirection est effectuée par une connexion aux serveurs spécifiés dans les paramètres `Serveur` et `Port` du groupe `ALTERNATE_SERVER_LIST`. Chaque serveur de la liste alternative est essayé au maximum le nombre de fois défini dans `maxAcrRetries` avant de passer au suivant et ce, jusqu'à épuisement de la liste ou l'établissement d'une connexion. Par exemple, si la liste de serveurs alternatifs est définie comme `(s3,s1,s2)` et que `maxAcrRetries` a la valeur 3, l'ordre de redirection sera alors `(s3,s3,s3) (s1,s1,s1) (s2,s2,s2)`. Le paramètre `acrRetryInterval` est appliqué entre chaque tentative de connexion individuelle (pour un total de 8 fois dans cet exemple).

Procédure

Pour spécifier des paramètres d'affinités client dans le fichier de configuration db2dsdriver, procédez comme suit :

1. Dans le fichier de configuration db2dsdriver sous le groupe ACR, déclarez la liste de serveurs alternatifs en spécifiant des entrées pour ALTERNATE_SERVER_LIST. Au sein de cette liste, vous devez spécifier le numéro de port et le nom d'hôte de chaque serveur. Par exemple :

```
<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>
  </acr>
</database>
```

Cette liste a pour seul effet de déclarer les serveurs alternatifs ; elle n'entraîne aucune modification de comportement.

2. Sous le groupe ACR, identifiez l'ordre des serveurs en spécifiant des entrées pour AFFINITY_LIST. Par exemple :

```

  <affinity_list>
    <list name="list1"
      serverorder="server1,server2,server3" >
    </list>
    <list name="list2"
      serverorder="server3,server2,server1" >
    </list>
  </affinity_list>
</acr>
```

Cette liste a pour seul effet d'identifier l'ordre des serveurs ; elle n'entraîne aucune modification de comportement.

3. Sous le groupe ACR, activez les affinités client en spécifiant l'un des sous-groupes suivants d'affinités client. La spécification d'un de ces sous-groupes force l'activation des affinités client. Tous les clients se connectant à cette base de données doivent être spécifiés dans l'un des sous-groupes CLIENT_AFFINITY. Si un client est introuvable dans l'un de ces sous-groupes, une erreur sera renvoyée lors de la tentative de connexion. Lorsqu'un sous-groupe CLIENT_AFFINITY est présent, la redirection automatique de client est implicitement activée.

- CLIENT_AFFINITY_DEFINED

Stipule un mappage spécifique du nom d'hôte du client à un élément AFFINITY_LIST spécifique. Le nom d'hôte du client est reconnu automatiquement et mis en correspondance avec l'entrée du fichier de configuration db2dsdriver pour établir la liste d'affinités. Par exemple :

```
.
.
.
<client_affinity_defined>
  <!-- cette section comporte des affinités spécifiques définies
  -->
  <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

Spécifie une permutation circulaire au sein de la liste ALTERNATE_SERVER_LIST. Cette affectation définit comme index client (basé zéro) le serveur de départ dans la liste CLIENT_AFFINITY_ROUNDROBIN, modulo le nombre de serveurs dans ALTERNATE_SERVER_LIST. Par exemple :

```
.
.
.
<client_affinity_roundrobin>
  <!-- roundrobin sélectionne le serveur de départ comme
  numéro d'index client dans cette section (basé 0)
  modulo le nombre de serveurs.
  -->
  <client name="client3"
    hostname="appsrv3.svl.ibm.com" >
    <!-- cette entrée porte l'index 0, modulo 3, et renvoie donc :
    server1, server2, server3
    -->
  </client>
  <client name="client4"
    hostname="appsrv4.svl.ibm.com" >
    <!-- cette entrée porte l'index 1, modulo 3, et renvoie donc :
    server2, server3, server1
    -->
  </client>
</client_affinity_roundrobin>
.
.
.
```

Résultats

Les considérations et règles de validation suivantes s'appliquent au traitement des affinités client :

- Lorsque CLIENT_AFFINITY est activé (c'est à dire lorsque un ou plusieurs sous-groupes CLIENT_AFFINITY sont présents dans le fichier de configuration db2dsdriver), la valeur par défaut de maxAcrRetries est 3. La plage de valeurs valides va de 0 à MAX_INT. Si la valeur spécifiée est 0, le traitement de redirection automatique de client opère comme si maxAcrRetries avait été défini à 1.
- La valeur par défaut de acrRetryInterval est 0. La plage de valeurs valides va de 0 à MAX_INT. Lorsque cette valeur est 0, aucun délai n'intervient entre les tentatives de connexion ou les tentatives de redirection automatique du client dans le pilote.
- Le nombre total de serveurs alternatifs éligibles pour un client donné ne peut pas dépasser 24. SQL1042N est renvoyé si le nombre de serveurs alternatifs éligibles est supérieur à 24.
- Les paramètres enableWLB et CLIENT_AFFINITY s'excluent mutuellement. Une erreur SQL5162N est renvoyée si enableWLB et CLIENT_AFFINITY ont tous deux été activés.
- Les attributs requis doivent être spécifiés dans les groupes ALTERNATE_SERVER_LIST, AFFINITY_LIST et CLIENT_AFFINITY, faute de quoi SQL5163N est renvoyée.
- Si CLIENT_AFFINITY est activé, la liste ALTERNATE_SERVER_LIST ne doit pas être vide, faute de quoi SQL5164N est renvoyée.
- Si CLIENT_AFFINITY est activé, le nom d'hôte du client doit déboucher sur une liste de serveurs alternatifs valide, faute de quoi SQL5164N est renvoyée. Par exemple, si le nom d'hôte du client est "xyz.domain.com" et que ce nom d'hôte ne peut être identifié dans aucun sous-groupe CLIENT_AFFINITY, SQL5164N est renvoyée. De plus, le nom d'hôte ne peut apparaître que dans un seul groupe CLIENT_AFFINITY.

Exemple

L'exemple suivant illustre une section de base de données dans le fichier de configuration db2dsdriver avec des sous-groupes CLIENT_AFFINITY spécifiés :

```
<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>
      <!-- cette section comporte des affinités spécifiques définies
      -->
      <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 sélectionne le serveur de départ comme
      numéro d'index client dans cette section (basé 0)
      modulo le nombre de serveurs.
      -->
      <client name="client3"
        hostname="appsrv3.svl.ibm.com" >
        <!-- cette entrée porte l'index 0, modulo 3, et renvoie donc :
        server1, server2, server3
        -->
      </client>
      <client name="client4"
        hostname="appsrv4.svl.ibm.com" >
        <!-- cette entrée porte l'index 1, modulo 3, et renvoie donc :
        server2, server3, server1
        -->
      </client>
    </client_affinity_roundrobin>
  </acr>
</database>
```

Limitations pour l'utilisation du support du Sysplex client

Certaines limitations s'appliquent au support du Sysplex client disponible dans les clients de serveur de données IBM et dans les pilotes de serveur de données non Java.

- Limitations relatives à l'équilibrage de charge au niveau des transactions :
 - Le client doit se connecter via un protocole TCP/IP.
 - Pour les clients et les pilotes de serveur de données IBM qui utilisent un support de Sysplex, la liste des serveurs est mise en cache au sein du processus applicatif. Par conséquent, la liste est partagée uniquement par les connexions établies dans le cadre de ce processus afin d'effectuer l'équilibrage de charge de travail, à partir d'une ou de plusieurs unités d'exécution.
- Limitations relatives à la redirection automatique du client :
 - Le client doit se connecter via un protocole TCP/IP.
 - La fonction de reprise en ligne en continu sur la première instruction SQL d'une transaction n'est pas prise en charge si le serveur du Sysplex n'autorise pas la réutilisation du transport à la fin de la transaction précédente. Il existe une exception à cette règle si la réutilisation du transport n'est pas accordée à la suite de `KEEPDYNAMIC(YES)`.
 - La fonction de reprise en ligne en continu sur la première instruction SQL d'une transaction n'est pas prise en charge si le pilote de l'interface CLI doit gérer un historique de toutes les API appelées au cours d'opérations précédentes afin de réexécuter l'instruction SQL. Ceci peut se produire lors de la spécification de données pendant des temps d'exécution, réalisation d'instructions SQL composées et utilisation d'entrée de tableau.
 - Si une fonction de reprise en ligne en continu se produit lorsque la validation automatique est activée, une instruction SQL risque d'être exécutée et validée plusieurs fois. Ceci peut se produire si l'événement qui déclenche la redirection automatique du client intervient après que la validation automatique initiale ait abouti. Pour éviter que ce comportement ne se produise, désactivez la fonction de validation automatique ou la fonction de reprise en ligne en continu.
 - La fonction de reprise en ligne en continu n'est pas disponible à l'intérieur d'une procédure mémorisée.
 - La fonction de reprise en ligne en continu n'est pas disponible dans des environnements fédérés.
- Support XA :
 - Les modèles de traitement du gestionnaire de transactions suivants ne sont pas pris en charge :
 - Transport double. Dans ce modèle, une transaction, sur un transport A, est liée à un membre à partir de `xa_start` jusqu'à `xa_end`, mais `xa_prepare` (en lecture seule), `xa_prepare` plus `xa_commit` ou `xa_rollback`, ou `xa_rollback` apparaît sur le transport B, résultant probablement d'un autre processus applicatif. Parmi les exemples de gestionnaires de transaction utilisant ce modèle, on trouve IBM WebSphere MQ et IBM Lotus Domino.
 - Multi-transport. Ce modèle implique l'utilisation de plusieurs transports à partir de plusieurs processus applicatifs, pour la même transaction. Parmi les exemples de gestionnaires de transactions utilisant ce modèle, on trouve BEA Tuxedo et IBM TXSeries Encina Monitor.
 - Pour les gestionnaires de transactions XA utilisant un modèle de traitement multi-transport, un serveur DB2 Connect intermédiaire est toujours requis.

Chapitre 20. Optimisation de DB2 Connect

Vous pouvez utiliser divers paramètres du fichier de configuration du gestionnaire de base de données pour régler DB2 Connect.

RQRIOBLK

Le paramètre **RQRIOBLK** définit la taille maximale des blocs d'entrée-sortie réseau. Une taille de bloc plus grande améliore les performances de requêtes plus grandes. La taille de bloc n'affecte généralement pas le temps de réponse des petites requêtes, telles qu'une requête d'une simple ligne de données.

Une taille de bloc plus grande requiert généralement davantage d'espace mémoire sur le serveur DB2 Connect. La taille de la partie active s'en trouve augmentée et peut engendrer une grande quantité de pagination sur de petites postes de travail.

Utilisez la taille de bloc DRDA par défaut (32767) si elle ne provoque pas trop de pagination à l'exécution de votre application. Autrement, réduisez la taille de bloc d'entrée-sortie jusqu'à ce qu'il n'y ait aucune pagination. Une fois la pagination démarrée, une diminution sensible des performances se produira. Utilisez les outils de contrôle des performances (tels que l'outil `vmstat` pour les systèmes d'exploitation Linux et UNIX) afin de déterminer si la pagination a lieu sur votre système.

DIR_CACHE

Le paramètre **DIR_CACHE_CACHE** détermine si les renseignements répertoire sont placés dans la mémoire cache. Avec la mise en cache (`DIR_CACHE=YES`), les fichiers répertoire sont lus et placés dans la mémoire cache afin de diminuer le temps système lié à la création d'une structure de répertoire interne et à la lecture des fichiers répertoire chaque fois qu'une connexion est établie.

Sans mise en cache (`DIR_CACHE=NO`), lorsque vous vous connectez à une base de données, le répertoire approprié est lu à partir d'un disque et la recherche est effectuée. Une fois les entrées de requête trouvées, tout l'espace mémoire dédié aux recherches de répertoire est libéré.

Avec la mise en cache, une mémoire cache de répertoire partagé est générée lors du traitement `db2start` et libérée lorsque DB2 s'arrête. Cette mémoire cache est utilisée par tous les processus du serveur DB2 (`db2agent`). Aussi, une mémoire cache de répertoire d'application privée est générée lorsqu'une application effectue sa première connexion à une base de données et libérée lorsque l'application se termine.

Chaque mémoire cache fournit une image du répertoire système des bases de données, du répertoire des services de connexion à la base de données et du répertoire des noeuds. La mémoire cache réduit les coûts de connexion en éliminant les entrées-sorties du fichier de répertoire et en limitant les recherches dans le répertoire.

Si un répertoire cache est mis à jour, les modifications ne sont pas propagées immédiatement aux mémoires cache. Si une entrée de répertoire n'est pas détectée dans une mémoire cache, le répertoire d'origine est parcouru.

La mise en cache augmente la quantité de mémoire privée requise par une application. Sans la mise en cache, l'espace mémoire est uniquement requis lorsqu'une recherche de répertoire est effectuée. L'utilisation générale de la mémoire partagée par DB2 augmente légèrement car les renseignements répertoire partagés entre les agents de base de données sont déplacés dans la mémoire partagée. La taille de l'espace mémoire requis pour une mémoire cache dépend du nombre d'entrées définies dans chaque répertoire.

NUMDB

Le comportement de DB2 Connect n'était pas affecté par le paramètre de configuration **NUMDB** dans les versions précédentes, ce qui n'est plus le cas à partir de la version 8. Ce paramètre indique le nombre maximal de bases de données auxquelles les clients peuvent se connecter via le serveur DB2 Connect. Plus spécifiquement, il s'agit du nombre maximal d'alias de base de données différents pouvant être catalogués sur le serveur DB2 Connect.

Autres paramètres DB2 Connect

Les paramètres **AGENTPRI** et **MAXAGENTS** sont obsolètes dans la version 9.5.

Les commandes prévues pour la mise à jour de la valeur du paramètre **MAXAGENTS** restent exécutables afin de ne pas perturber les applications existantes, mais les valeurs sont ignorées. Le nom du paramètre n'apparaîtra dans aucune liste de configuration. Jusqu'ici, le nombre total d'agents pouvant être créés sur une partition DB2 donnée était déterminé par le paramètre de configuration **MAXAGENTS**. Vous avez désormais la possibilité d'automatiser la configuration des agents.

Par défaut, l'option **AUTOMATIC** et une valeur de 100 seront associées au paramètre **NUM_POOLAGENTS**. Pour le paramètre **MAX_COORDAGENTS**, il s'agira de l'option **AUTOMATIC** et de la valeur 200.

Pour envoyer les identifiants comptables des applications client vers le serveur DB2 Connect, utilisez les moyens propres à l'API pour définir des informations statistiques. Les procédés propres à l'API s'exécutent plus rapidement que la configuration de la variable d'environnement **DB2ACCOUNT**.

IBM Data Server Driver for JDBC and SQLJ

`com.ibm.db2.jcc.DB2BaseDataSource.clientAccountingInformation`
(propriété)

IBM Data Server Provider pour .NET (Data Server Provider for .NET)

`DB2Connection.ClientAccountingInformation` (propriété)

CLI/ODBC

Mot clé de configuration `ClientAcctStr` CLI/ODBC

SQL imbriqué (C, C++ et COBOL)

`sqlsact` (fonction)

Si vous n'avez pas besoin d'un fichier de mappage **SQLCODE** personnalisé, vous pouvez améliorer les performances en utilisant le mappage **SQLCODE** par défaut ou en désactivant le mappage **SQLCODE**. Le fichier de mappage par défaut est imbriqué dans la bibliothèque DB2 Connect et un fichier de mappage personnalisé doit être lu à partir du disque, ce qui affecte les performances.

Optimisation de la base de données hôte

Les performances du système seront affectées par celles du serveur de base de données grand système IBM. Les systèmes de gestion de base de données possèdent leurs propres fonctions de performances. Les optimiseurs SQL des divers systèmes peuvent, par exemple, avoir un comportement différent avec la même application. Pour plus d'informations, consultez la documentation relative aux performances système de votre serveur de base de données grand système IBM.

Vous pourrez peut-être améliorer les performances en utilisant les options de définition d'accès de lecture non validée, lorsqu'elles sont disponibles, pour contourner la journalisation.

Remarque : Lorsque vous utilisez la lecture non validée, les données non consignées peuvent uniquement être lues et ne peuvent être mises à jour, et ce uniquement si le blocage est défini sur ALL.

En fonction de votre serveur d'applications et de la granularité du verrouillage, le niveau d'isolement utilisé pour une requête ou une application peut avoir un impact conséquent sur les performances. La base de données doit posséder le niveau approprié de normalisation, une utilisation des index efficace et une allocation adéquate de l'espace mémoire de la base de données. Les performances peuvent également être affectées par les types de données que vous utilisez (voir sections suivantes).

Considérations d'optimisation réseau

Le meilleur procédé pour améliorer les performances globales dans un environnement de base de données réparti consiste à éliminer les retards du réseau. Les administrateurs réseau considèrent généralement qu'un réseau est plus efficace lorsqu'il rassemble autant de données que possible entre les transmissions. Cette approche ne fonctionne pas pour les applications telles que les bases de données réparties car elle crée des retards dans le réseau. L'utilisateur final ne constate pas l'efficacité du réseau, uniquement les retards.

La plupart des périphériques réseau possèdent des paramètres de retard et leurs valeurs par défaut sont généralement définies sur des valeurs loin d'être optimales pour les bases de données réparties. Afin d'améliorer les performances, vous devez localiser ces paramètres et, si possible, les définir sur zéro. En outre, vous devez vérifier que la taille de la mémoire tampon du périphérique est suffisante pour éviter toute retransmission suite à des pertes de données. Par exemple, les systèmes UNIX possèdent généralement une longueur de file d'attente de transmission ou de réception définie par défaut sur 32. Afin d'améliorer les performances, définissez la longueur de la file d'attente sur 150. Un paramètre correspondant dans les paramètres de contrôle de liaison de données est la longueur de la file d'attente de réception qui doit également être définie sur 150.

Le paramètre IOBUF est généralement défini sur une valeur trop faible sur la plupart des sites. Elle est également définie sur 500. Mais l'expérience a démontré avec le temps qu'une valeur de 3992 est plus adaptée si vous déplacez de grandes quantités de données, notamment si vous utilisez des connexions par canaux, telles que les connexions ESCON ou 3172.

Sur les systèmes locaux, les tailles des fenêtres de réception et de transmission DLC (contrôle de liaison de données) ou LLC (contrôle de liaison logique) peuvent

affecter considérablement les performances. La valeur envoyée doit être définie sur une valeur supérieure ou égale à 7 ; pour la plupart des configurations, une valeur égale ou inférieure à 4 engendre des résultats inférieurs.

Si vous utilisez Ethernet, définissez la taille de segment TCP sur 1500 octets. Sur un réseau en anneau à jeton ou FDDI, vous devez définir cette valeur sur 4400 octets et si vous utilisez un adaptateur ESCON avec le protocole TCP/IP, la taille de segment doit toujours être définie sur 4096.

En ce qui concerne les réseaux TCP/IP, les tailles de mémoire tampon d'envoi et de réception TCP doivent être définies sur des valeurs supérieures à 32768. Une valeur de 65536 fournira généralement les meilleures performances.

Remarque : L'établissement d'une connexion de la passerelle au serveur (connexion sortante) consomme davantage de ressources que l'établissement d'une connexion d'un client à la passerelle (connexion entrante). Dans un environnement dans lequel des milliers de clients se connectent fréquemment au serveur via une passerelle et s'en déconnectent, un temps de traitement considérable est consacré à l'établissement de connexions sortantes. DB2 Connect offre le regroupement des connexions via le protocole TCP/IP. Lorsqu'un client demande à se déconnecter du serveur, la passerelle supprime la connexion entrante du client, mais conserve la connexion sortante vers le serveur dans un regroupement. Lorsqu'un nouveau client arrive sur la passerelle et demande une connexion, la passerelle fournit une connexion existante du regroupement, ce qui réduit les temps de connexion globaux et permet d'éviter de consommer un grand nombre de ressources de l'unité centrale.

Pour consulter un récapitulatif des méthodes de réglage des performances réseau, voir tableau 19.

Tableau 19. Méthodes de réglage des performances réseau

Eléments à rechercher	Exemple	Paramètre	Remarques
Retards volontaires	Paramètre de retard sur les périphériques réseau	Défini sur 0.	Les valeurs par défaut sont généralement supérieures.
Mémoires tampon	Paramètre IOBUF	Défini jusqu'à 3992.	Particulièrement utile pour l'adaptateur ESCON ou d'autres adaptateurs de canal.
Mémoires tampon	RUSIZE	Taille optimale de 4096.	La définition des paramètres RUSIZE et RQRIOBLK sur la même taille peut engendrer les meilleures performances.
Mémoires tampon	Régulation	VPACING, PACING et Mode Profiles doivent être définis sur 63.	Utilisez la régulation adaptative, si applicable.
Paramètres de l'adaptateur	Longueur de la file d'attente de transmission/réception	La valeur recommandée est 150.	La valeur par défaut est généralement 32.

Tableau 19. Méthodes de réglage des performances réseau (suite)

Eléments à rechercher	Exemple	Paramètre	Remarques
Paramètres TCP	Tailles de segment	1500 avec Ethernet, 4400 avec anneau à jeton et interface optique FDDI	Les adaptateurs ESCON utilisés pour le protocole TCP/IP doivent toujours être définis sur 4096.
Paramètres TCP	Tailles d'espace d'envoi/de réception	Doit être de 64 K pour les deux.	La valeur par défaut est de seulement 8192 pour Windows. Peut être défini dans le registre Windows.

Conflit de ressources système

Les performances peuvent être moindres si de nombreuses tâches du système tentent d'obtenir des ressources système. Tenez compte des questions suivantes :

- L'unité centrale est-elle saturée ? Pensez à mettre le système à niveau, à réduire la charge de travail du système et à régler le système de manière à réduire le temps système de traitement.
- L'espace mémoire est-il suralloué ? Pensez à mettre l'espace mémoire à niveau, à réduire la charge de travail du système et à régler le système de manière à réduire la partie active de l'espace mémoire.
- Le contrôleur de la carte/de communication est-il surchargé ? Pensez à mettre le réseau à niveau ou à apparier les cartes de réseau en anneau à jeton.
- L'un des sous-systèmes est-il surchargé. Si oui, s'agit-il du sous-système sur le chemin d'accès aux données ?
- Des tâches et des processus inutiles sont-ils en cours d'exécution sur le système ? La conduite générale consiste à ne pas configurer ou à démarrer des services sauf s'ils sont utilisés régulièrement car ils gaspillent les ressources système.
- Certains processus ou certaines tâches utilisent-ils/elles la majeure partie des ressources ? Peuvent-ils/elles être arrêté(e)s ? Leur priorité peut-elle être diminuée ? Peuvent-ils/elles être redéfini(e)s afin d'utiliser moins de ressources ?

Résolution des incidents de performances de DB2 Connect

Si les utilisateurs de DB2 Connect constatent des temps de réponse prolongés lors de requêtes volumineuses auprès de serveurs grand système IBM, les problèmes de performances peuvent provenir des domaines suivants :

1. Pour les requêtes qui renvoient de grands blocs de données du serveur grand système IBM (généralement 32 Ko de données et plus), vérifiez que la valeur du paramètre RQRIOLBK du gestionnaire de configuration est définie à 32767. Effectuez cette opération à l'aide de l'interpréteur de commandes (CLP) en procédant comme suit :


```
db2 update database manager configuration using RQRIOLBK 32767
```
2. Vérifiez que la taille maximale de RU définie dans la définition du mode IBMRDB est définie sur une valeur convenable. Il est recommandé de ne pas définir la taille sur une valeur inférieure à 4 K pour les connexions à l'aide de

matériel en anneau à jeton. Pour les connexions utilisant des matériels Ethernet, notez que la taille de trame maximale Ethernet de 1536 octets peut être un facteur restrictif.

Optimisation de DB2 for z/OS

Vous pouvez optimiser le traitement d'unités d'exécution inactives dans z/OS. Dans la version 5, 25.000 clients connectés concurrents sont autorisés. Dans tous les cas de figure, le nombre maximal de clients pouvant être actifs de manière concurrente est cependant de 1999. Chaque client de poste de travail peut rester connecté lorsqu'il est inactif, son unité d'exécution est placée dans une chaîne inactive à chaque validation.

Les paramètres DSNZPARM CMTSTAT, CONDBAT et MAXDBAT affectent le traitement par unité d'exécution. Pour optimiser les performances, définissez CMTSTAT sur INACTIVE, ajustez CONDBAT sur le nombre maximal de DBAT autorisés offrant de bonnes performances et MAXDBAT sur le nombre maximal acceptable de DBAT actifs.

Augmentation des débits de transfert des données de DB2 Connect

Outre la capacité de bloquer des lignes d'un ensemble de résultats de requêtes, DB2 for z/OS peut également renvoyer des multiples, tels que des blocs de requêtes, en réponse à une requête OPEN ou FETCH vers un client distant, comme DB2 Connect. Au lieu d'envoyer des requêtes successives vers le serveur DB2 for z/OS en lui réclamant un bloc de données de lignes à la fois, le client peut désormais demander à ce qu'il lui renvoie un nombre spécifique de blocs, en plus de celui qu'il renvoie toujours. Ces blocs de requêtes sont appelées "blocs de requêtes supplémentaires".

Cette nouvelle fonction permet au client de réduire le nombre d'allers et retours de la ligne de réseau qui représentent un coût majeur dans les performances réseau. La diminution du nombre de demandes de blocs de requêtes envoyées par le client au serveur se traduit par une augmentation significative des performances. Cette amélioration des performances est due au fait que le basculement entre l'envoi et la réception est une opération coûteuse en termes de performances. DB2 Connect peut désormais exploiter cette amélioration des performances en demandant des blocs de requêtes supplémentaires à un serveur DB2 for z/OS par défaut.

Pour tirer entièrement partie du renvoi de blocs de requêtes supplémentaires (pouvant chacun posséder une longueur de 32 K) pour le protocole réseau préféré TCP/IP, les extensions de mise à l'échelle des fenêtres ont été activées comme ayant été structurées sous RFC-1323 dans DB2 Connect. Cette fonction permet au protocole TCP/IP de régler dynamiquement et efficacement les tailles des fenêtres d'envoi et de réception pour les adapter à d'éventuelles grandes quantités de données renvoyées au moyen de blocs de requêtes supplémentaires.

Bloc de requête supplémentaire

La prise en charge de blocs de requête supplémentaires sur les serveurs DB2 for z/OS version 7, ou ultérieure, est configurée via le paramètre EXTRA BLOCKS SRV du panneau d'installation DDF de DB2. Cette prise en charge est configurée par le contrôle du nombre maximal de blocs de requêtes supplémentaires que DB2 peut renvoyer à un client pour une requête. Vous pouvez définir ce paramètre sur une valeur comprise entre 0 et 100. La définition de la valeur de ce paramètre sur 0 entraîne la désactivation du renvoi de blocs de requête supplémentaires. La

valeur par défaut (100) doit toujours être utilisée pour optimiser cette fonction, car elle bloque toute idiosyncrasie du réseau qui pourrait rendre cette valeur moins appropriée.

Côté client, où l'application accède à DB2 for z/OS soit directement via une installation serveur DB2 Connect colocalisée, soit via une installation serveur DB2 Connect séparée, divers procédés sont possibles pour activer la prise en charge DB2 Connect correspondante à l'aide du curseur ou d'une instruction :

- L'utilisation d'une taille d'ensemble de lignes de requête pour un curseur
- L'utilisation de la clause 'OPTIMIZE for N ROWS' dans l'instruction Select associée à un curseur
- L'utilisation de la clause 'FETCH FIRST N ROWS ONLY' dans l'instruction Select associée à un curseur

DB2 Connect peut activer la prise en charge des blocs de requête supplémentaires à l'aide de diverses API SQL :

SQL imbriqué

- L'utilisateur peut appeler la prise en charge de blocs de requête supplémentaires pour une requête en spécifiant les clauses 'OPTIMIZE for N ROWS' et/ou 'FETCH FIRST N ROWS ONLY' dans l'instruction Select.
- A l'aide de la clause 'OPTIMIZE for N ROWS', DB2 for z/OS tente de bloquer le nombre souhaité de lignes à renvoyer à DB2 Connect, en fonction du paramètre d'installation EXTRA BLOCKS SRV DDF. L'application peut choisir d'extraire plus de N lignes puisque DB2 for z/OS ne limite pas à N le nombre de lignes pouvant être renvoyées dans l'ensemble de résultats de la requête.
- La clause 'FETCH FIRST N ROWS ONLY' opère de manière similaire, si ce n'est que l'ensemble de résultats de la requête est limité à N lignes par DB2 for z/OS. L'extraction de plus de N lignes engendre l'apparition d'un code SQL +100 (fin de données).

CLI/ODBC

- L'utilisateur peut appeler la prise en charge de blocs de requêtes supplémentaires pour une requête via son attribut d'état SQL_MAX_ROWS.
- La clause 'FETCH FIRST N ROWS ONLY' est utilisée à la place pour un serveur DB2 for z/OS 7.1, ou version ultérieure.
 - Pour la version 7, l'ensemble de résultats de la requête est limité à N lignes par DB2 for z/OS. L'extraction de plus de N lignes engendre l'apparition d'un code SQL_NO_DATA_FOUND.
 - Pour la version 8 ou version ultérieure, l'interface CLI garantit que seules les premières lignes sont renvoyées à l'application via le client Gestionnaire de curseur.

JDBC L'utilisateur peut appeler la prise en charge de blocs de requêtes supplémentaires pour une requête via la méthode setMaxRows. Comme pour l'activation CLI/ODBC, DB2 Connect balise la clause 'OPTIMIZE for N ROWS' pour un serveur DB2 for z/OS 6.x. DB2 Connect balise également la clause 'FETCH FIRST N ROWS ONLY' pour un serveur DB2 for z/OS 7.1 ou version ultérieure.

Mise à l'échelle des fenêtres RFC-1323

La mise à l'échelle des fenêtres est prise en charge sur toutes les plateformes Windows, Linux, et UNIX prenant en charge les extensions RFC-1323 pour le protocole TCP/IP. Vous pouvez activer cette fonction sous DB2 pour Windows, Linux ou UNIX à l'aide de la variable de registre DB2 DB2SORCVBUF. Pour activer la mise à l'échelle des fenêtres, la variable de registre doit être définie sur une valeur supérieure à 64 K. Par exemple, sous DB2 pour Windows, Linux, ou UNIX, vous pouvez exécuter `db2set DB2SORCVBUF =65537`.

Les tailles de mémoire tampon d'envoi et de réception maximales dépendent du système d'exploitation spécifique. Pour garantir que les tailles de mémoire tampon configurées ont été acceptées, l'utilisateur peut définir le paramètre de configuration du gestionnaire de base de données DIAGLEVEL sur 4 (informationnel) et parcourir le fichier journal de notification de l'administration à la recherche de messages.

Pour que la mise à l'échelle des fenêtres prennent effet, elle doit être activée aux deux extrémités de la connexion, au niveau du poste de travail et de l'hôte, directement via la pile de protocole TCP/IP du système d'exploitation ou indirectement via le produit DB2. Par exemple, pour DB2 for z/OS, la mise à l'échelle des fenêtres ne peut être activée actuellement via le système d'exploitation qu'en définissant TCPRCVBUFRSIZE à une valeur supérieure à 64 K. Si vous utilisez un client IBM Data Server distant pour accéder à une base de données grand système IBM DB2 via un poste de travail de serveur DB2 Connect, vous pouvez également activer la mise à l'échelle sur le client. De même, vous pouvez activer la mise à l'échelle des fenêtres entre un client IBM Data Server distant et un poste de travail de serveur DB2 lorsque aucune base de données grand système IBM DB2 n'est impliquée.

Alors que la mise à l'échelle des fenêtres est conçue pour améliorer les performances réseau, il est important de noter que l'amélioration des performances réseau prévue ne se concrétise pas toujours. L'interaction entre les facteurs tels que la taille de la trame utilisée par l'adaptateur Ethernet ou LAN en anneau à jeton, la taille MTU IP ainsi que d'autres paramètres au niveau des routeurs à travers la liaison peut également engendrer une dégradation des performances lorsque la mise à l'échelle des fenêtres a été activée. Par conséquent, la mise à l'échelle des fenêtres est désactivée par défaut lorsque les deux mémoires tampon d'envoi et de réception sont définis sur 64 K.

Mesurez l'impact de l'activation de la mise à l'échelle des fenêtres et effectuez tout ajustement nécessaire sur le réseau. Une présentation de l'optimisation du réseau pour des performances réseau améliorées est disponible sur le site <http://www.networking.ibm.com/>.

Conversion de données sur l'hôte

Lors d'un transfert d'informations entre des environnements différents (tels que systèmes d'exploitation Intel [Windows], IEEE [Linux et UNIX], System z [VM, VSE, z/OS], IBM Power Systems [IBM i]), des types de données numériques (tels que les décimales, les entiers ou les nombres en virgule flottante) auront peut être besoin d'être convertis. Cette conversion peut affecter les performances.

Le coût en termes d'utilisation de l'unité centrale de la conversion des données de caractères mono-octet, est généralement inférieur à celui de la conversion des données numériques (lorsque la conversion des données est nécessaire).

Le coût de la conversion de données de DATE/TIME/TIMESTAMP est pratiquement identique à celui de CHAR mono-octet. La conversion de virgules flottantes FLOATING est le procédé le plus onéreux. Le concepteur de l'application peut souhaiter bénéficier de ces états de fait lorsqu'il conçoit une application basée sur DB2 Connect.

Si une table de base de données possède une colonne définie sur 'FOR BIT DATA', les données de type caractères transférées entre l'application et la base de données ne requièrent aucune conversion. Cette solution peut être utilisée lorsque vous archivez des données sur le serveur de base de données grand système IBM.

Types de données pour les données de type caractères

Les données de type caractères peuvent posséder le type de données CHAR ou VARCHAR. Le type de données le plus efficace dépend de la longueur moyenne des données dans le champ :

- Si la taille des données actuelles varie fréquemment, le type VARCHAR sera le plus efficace car le type CHAR ajoute des caractères blancs supplémentaires pour remplir le champ. Ces caractères blancs doivent être transmis via le réseau comme n'importe quel autre caractère.
- Si la taille des données actuelles ne varie pas beaucoup, le type CHAR sera le plus efficace car chaque champ VARCHAR possède des informations d'une longueur de quelques octets devant être transmises.

Matériel réseau

Les remarques suivantes concernent le matériel :

- Vitesse du réseau ou du support de transmission

Les performances s'améliorent lorsque vous utilisez un support de transmission plus rapide. Par exemple, les valeurs suivantes sont des vitesses de transfert de données brutes :

De canal à canal (fibre optique)

4,0 Mo/s

Réseau LAN de 16 Mbps

2,0 Mo/s

De canal à canal (classique)

1,0 Mo/s

Réseau LAN de 4 Mbps

0,5 Mo/s

Multiplex T1 à grande vitesse (1,544 Mbps)

0,193 Mo/s

Ligne téléphonique distante rapide de 56 Kbps

0,007 Mo/s

Modem 19,6 Kbps

0,002 Mo/s

Modem 9600 bps

0,001 Mo/s

Le débit du transfert de données est limité par le support de transmission le plus lent sur le chemin d'accès au serveur de base de données grand système IBM.

- L'adaptateur de réseau ou le contrôleur de communication
Planifiez soigneusement l'utilisation de l'espace mémoire de l'adaptateur de réseau ou du contrôleur de communication. En outre, travaillez avec un spécialiste du réseau afin de vérifier que le contrôleur possède les capacités de gérer le trafic supplémentaire généré par DB2 Connect.
- Topologie de réseau
Si les données sont transférées de réseau LAN en réseau LAN, et d'un réseau à un autre, prenez compte du temps de trajet. Les ponts, les routeurs et les passerelles augmentent le temps écoulé. Par exemple, la diminution du nombre de ponts traversés réduit le nombre de tronçons requis par chaque requête.
La distance physique entre les noeuds doit également être prise en compte. Même si un message est transféré par satellite, le temps de transfert est limité par la vitesse de la lumière ($3 * 10^{**8}$ m/s) et la distance de rotation entre l'expéditeur et le récepteur.
- Trafic réseau
Si la bande passante du réseau a été entièrement utilisée, le temps de réponse et la vitesse de transfert de données d'une seule application diminueront.
Une surcharge peut se produire dans le réseau lorsque des données s'accumulent dans un ancien programme de contrôle de réseau avec une mémoire tampon de petite taille.
- Fiabilité du réseau
Si le taux d'erreurs du réseau est élevé, le rendement du réseau diminue ce qui entraîne une diminution des performances en raison de la retransmission des données.

Chapitre 21. Optimisation des performances d'applications CLI/ODBC

CLI/ODBC est une interface de programme d'application SQL qui peut être appelée par vos applications de base de données. Les fonctions CLI invoquent des procédures mémorisées DB2 qui, à leur tour, accèdent à des tables du catalogue système.

Certaines applications utilisent des API ODBC pour rassembler des informations sur les métadonnées utilisées pour des traitements ultérieurs. Les dix appels d'API de métadonnées pouvant être effectués sont les suivants :

- SQLTables
- SQLColumns
- SQLSpecialcolumns
- SQLStatistics
- SQLPrimarykeys
- SQLForeignkeys
- SQLTablePrivileges
- SQLColumnPrivileges
- SQLProcedures
- SQLProcedureColumns

Certaines applications CLI/ODBC utilisant les API de métadonnées répertoriées ci-dessus peuvent requérir tous les objets au sein de la base de données. Par exemple, un appel SQLTables demande des métadonnées pour toutes les tables de la base de données. Sur des systèmes plus grands, ces requêtes peuvent engendrer un trafic réseau important et consommer une quantité considérable de temps et de ressources du serveur.

Plusieurs mots clés d'initialisation CLI/ODBC peuvent être utilisés pour limiter la quantité de données renvoyée par les appels initiaux de l'API lors de l'étape de rassemblement d'informations une fois la base de données connectée. Ces mots clés peuvent être définis via :

1. L'édition manuelle dans le fichier db2cli.ini.
2. La modification des paramètres ODBC/CLI de la base de données à l'aide de l'Assistant de configuration client (sur les plateformes qui le prennent en charge).
3. La mise à jour de la configuration CLI de la base de données à l'aide de l'interface de ligne de commande DBA.

Les mots clés sont :

- DBName
- TableType
- SchemaList
- SysSchema
- GrantorList
- GranteeList

Partie 5. Identification des incidents

Chapitre 22. Identification et résolution des incidents DB2 Connect

L'environnement DB2 Connect implique divers logiciels et matériels et diverses communications. La meilleure approche lors d'une procédure d'identification des incidents consiste à analyser les données disponibles puis à agir en conséquence pour tenter de localiser l'origine de l'erreur.

Après avoir rassemblé les informations pertinentes et selon votre sélection de la rubrique applicable, passez à la section référencée.

Collecte d'informations pertinentes

L'identification des incidents englobe la réduction de la portée de l'incident et la recherche des causes probables. Le point de départ idéal consiste à rassembler des informations adéquates et à déterminer vos connaissances, les données qui n'ont pas été rassemblées et les chemins d'accès que vous pouvez supprimer. Vous devez au minimum répondre aux questions suivantes.

- La connexion initiale était-elle fructueuse ?
- Le matériel fonctionne-t-il correctement ?
- Les chemins de communication sont-ils opérationnels ?
- Des modifications de réseau de communication ont-elles eu lieu qui rendraient les entrées de répertoire précédentes non valides ?
- La base de données a-t-elle été lancée ?
- L'interruption de communication est-elle survenue entre un ou plusieurs clients et DB2 Connect (passerelle) ou entre la passerelle DB2 Connect et le serveur de base de données grand système IBM ou entre DB2 Connect Personal Edition et le serveur de base de données grand système IBM ?
- Que vous pouvez déduire du contenu du message et des jetons renvoyés dans le message ?
- L'utilisation des outils de diagnostic tels que db2trc, db2pd ou db2support vous a-t-elle été d'une aide quelconque pour l'instant ?
- D'autres machines effectuant des tâches similaires fonctionnent-elles correctement ?
- S'il s'agit d'une tâche distante, réussissez-vous à l'exécuter localement ?

Connexion initiale non aboutie

Passez en revue les questions suivantes et vérifiez que les étapes d'installation ont été respectées :

1. *Le processus d'installation s'est-il achevé correctement ?*
 - Tous les logiciels prérequis étaient-ils disponibles ?
 - L'espace mémoire et l'espace disque étaient-ils suffisants ?
 - Le support client distant a-t-il été installé ?
 - L'installation du logiciel de communications s'est-elle déroulée sans conditions d'erreur ?

2. *Pour les systèmes d'exploitation UNIX, une instance du produit a-t-elle été créée ?*
 - En tant que racine, avez-vous créé un utilisateur ou un groupe afin qu'il devienne le propriétaire d'instance et le groupe sysadm ?
3. *Le cas échéant, les informations sur la licence ont-elles été traitées avec succès ?*
 - Pour les systèmes d'exploitation UNIX, avez-vous édité le fichier nodelock et saisi le mot de passe fourni par IBM ?
4. *Les communications du serveur de base de données grand système IBM et le poste de travail ont-elles été configurées correctement ?*
 - Trois configurations doivent être prises en considération :
 - a. La configuration du serveur de base de données grand système IBM identifie le demandeur d'application auprès du serveur. Le système de gestion de base de données du serveur grand système IBM comportera des entrées de catalogue définissant le demandeur en termes d'emplacement, de protocole réseau et de sécurité.
 - b. La configuration du poste de travail DB2 Connect définit la population cliente auprès du serveur et le serveur grand système IBM auprès du client.
 - c. La configuration du poste de travail client doit comporter des noms définis pour le poste de travail et le protocole de communication.
 - L'identification des incidents liés à l'échec de l'établissement d'une connexion initiale englobe la vérification de la spécification de noms complets et corrects d'unités physiques ou la vérification de la bonne spécification du numéro de port et du nom d'hôte pour les connexions TCP/IP.
 - L'administrateur de base de données du serveur grand système IBM et les administrateurs réseau disposent chacun d'utilitaires de diagnostic des incidents.
5. *Détenez-vous les droits d'accès requis par le système de gestion de base de données du serveur grand système IBM pour utiliser la base de données du serveur grand système IBM ?*
 - Prenez compte des droits d'accès de l'utilisateur, des règles du qualifieur de table et des résultats anticipés.
6. *Lorsque vous tentez d'utiliser l'interpréteur de commandes (CLP) pour exécuter des instructions SQL sur un serveur de base de données grand système IBM, y parvenez-vous ?*
 - Avez-vous suivi la procédure de liaison de l'interpréteur de commandes au serveur de base de données grand système IBM ?

Incidents rencontrés après une connexion initiale

Les interrogations suivantes sont fournies en tant que point de départ afin de vous aider à réduire la portée de l'incident.

1. *Avez-vous rencontré des circonstances de fonctionnement spéciales ou inhabituelles ?*
 - S'agit-il d'une nouvelle application ?
 - Avez-vous utilisé de nouvelles procédures ?
 - Y a-t-il eu des modifications récentes qui peuvent affecter le système ? Par exemple, avez-vous modifié des produits ou des applications depuis la dernière exécution réussie de l'application ou du scénario ?
 - Pour les programmes d'application, quelle API avez-vous utilisé pour créer le programme ?

- D'autres applications utilisant le logiciel ou les API de communication ont-elles été exécutées sur le système de l'utilisateur ?
 - Un groupe de correctifs a-t-il été récemment installé ? Si l'incident se produit quand un utilisateur tente de se servir d'une fonction qui n'a plus été utilisée (ou chargée) sur son système d'exploitation depuis son installation, déterminez le groupe de correctifs IBM le plus récent et chargez-le après avoir installé la fonction.
2. *Cette erreur s'est-elle déjà produite auparavant ?*
 - Un procédé de résolution documenté relatif aux conditions d'erreur précédentes a-t-il été développé ?
 - Quels en étaient les participants et peuvent-ils évaluer l'impact d'une action éventuelle ?
 3. *Avez-vous songé à utiliser des commandes du logiciel de communication qui renvoient des informations sur le réseau ?*
 - Le protocole TCP/IP peut posséder des informations extraites de l'utilisation des commandes et démons TCP/IP.
 4. *Les informations renvoyées dans la SQLCA (zone de communication SQL) ont-elles une quelconque utilité ?*
 - Les procédures de gestion des incidents doivent inclure des étapes consistant à examiner le contenu des champs SQLCODE et SQLSTATE.
 - Les SQLSTATE permettent aux programmeurs d'application de tester des classes d'erreurs communes à la famille des bases de données DB2. Dans un réseau de bases de données relationnelles réparties, ce champ peut constituer une base commune.
 5. *START DBM a-t-il été exécuté en tant que serveur ?* En outre, vérifiez que la variable d'environnement DB2COMM est définie correctement pour les clients qui accèdent au serveur à distance.
 6. *D'autres machines effectuant la même tâche peuvent-elles se connecter au serveur correctement ?* Il se peut que le nombre maximal de clients tentant de se connecter au serveur ait été atteint. Si un autre client se déconnecte du serveur, le client qui n'arrivait pas à se connecter auparavant parvient-il désormais à se connecter ?
 7. *La machine possède-t-elle le bon adressage ?* Vérifiez que la machine est unique sur le réseau.
 8. *Lors de la connexion à distance, les bons droits d'accès ont-ils été attribués au client ?* La connexion à l'instance peut être fructueuse mais il se peut que l'autorisation n'ait pas été accordée au niveau de la base de données ou de la table.
 9. *S'agit-il de la première machine qui se connecte à une base de données éloignée ?* Dans les environnements répartis, des routeurs ou des ponts entre des réseaux peuvent bloquer la communication entre le client et le serveur. Par exemple, lorsque vous utilisez le protocole TCP/IP, vérifiez que vous pouvez exécuter la commande PING sur l'hôte distant.

Outils de diagnostic

Lorsque vous rencontrez un problème, vous pouvez procéder comme suit :

- Toutes les données de diagnostic y compris les fichiers de vidage, les fichiers de déroutement, les fichiers de notification et les journaux des erreurs se trouvent dans le chemin défini par le paramètre de configuration du gestionnaire de la base de données pour le chemin de répertoire des données de diagnostic (**diagpath**) :

Si la valeur de ce paramètre de configuration est null, les données de diagnostic sont écrites dans l'un des répertoires ou dossiers suivants :

- Pour les environnements Linux et UNIX : `INSTHOME/sqllib/db2dump`, où `INSTHOME` est le répertoire initial de l'instance.
- Pour les environnements Windows pris en charge :
 - Si la variable d'environnement **DB2INSTPROF** n'est pas définie, `x:\SQLLIB\DB2INSTANCE` est alors utilisé, où `x:\SQLLIB` désigne l'unité et le répertoire indiqué dans la variable de registre **DB2PATH**, et la valeur de **DB2INSTANCE** a le nom de l'instance.

Remarque : Le répertoire ne doit pas obligatoirement être appelé `SQLLIB`.

- Si la variable d'environnement **DB2INSTPROF** est définie, `x:\DB2INSTPROF\DB2INSTANCE` est alors utilisé, où **DB2INSTPROF** est le nom du répertoire de profils d'instance et **DB2INSTANCE** est le nom de l'instance (par défaut, la valeur de **DB2INSTDEF** dans les systèmes d'exploitation Windows 32 bits).
- Pour les systèmes d'exploitation Windows, vous pouvez utiliser l'Observateur d'événements pour visualiser le journal de notification de l'administration.
- Les outils de diagnostic disponibles comprennent **db2trc**, **db2pd**, **db2support** et **db2diag**
- Pour les systèmes d'exploitation Linux et UNIX, la commande **ps** renvoie les informations relatives à l'état des processus actifs à la sortie standard.
- Pour les systèmes d'exploitation UNIX, le fichier `core` créé dans le répertoire courant lorsque des erreurs graves se produisent. Il contient une image mémoire du processus terminé et peut être utilisé pour déterminer la fonction à l'origine de l'erreur.

Chapitre 23. Traces DB2 dans DB2 Connect

Le traçage des actions et des opérations au fur et à mesure de leur déroulement dans votre environnement permet d'obtenir des informations utiles pour l'identification et la résolution d'un incident. Vous pouvez obtenir, vider et formater une trace exécutée dans le serveur de base de données DB2. La fonction de trace est fournie avec le serveur de base de données DB2.

Obtention d'une trace DB2 avec db2trc

La commande db2trc gère la fonction de trace fournie avec DB2. La fonction de trace enregistre les informations sur les opérations et les formate de façon à les rendre lisibles.

Rappelez-vous que l'exécution d'une trace consomme du temps système supplémentaire et que l'activation de la fonction de trace peut donc affecter les performances de votre système.

En général, les équipes de support et de développement de DB2 utilisent les traces à des fins d'identification et de résolution des incidents. Vous pouvez exécuter une trace pour obtenir des informations sur un incident que vous analysez, mais son utilité reste restreinte si vous ne connaissez pas le code source de DB2.

Néanmoins, il est important de savoir activer correctement le traçage et de prendre des clichés des fichiers de trace au cas où il vous serait demandé de les fournir.

Remarque : Vous devez posséder les droits d'accès SYSADM, SYSCTRL ou SYSMAINT pour pouvoir utiliser db2trc

Pour vous faire une idée générale des options disponibles, exécutez la commande db2trc sans indiquer de paramètres :

```
C:\>db2trc
Syntaxe : db2trc (chg|clr|dmp|flw|fmt|inf|off|on) options
```

Pour plus d'informations sur un paramètre de la commande db2trc, utilisez l'option -u. Par exemple, pour afficher des informations supplémentaires sur l'activation de la trace, exécutez la commande suivante :

```
db2trc on -u
```

Celle-ci vous fournit des informations sur toutes les options supplémentaires (appelées des "fonctions") que vous pouvez indiquer lorsque vous activez une trace DB2.

Lorsque vous activez une trace, l'option la plus importante est -L. Elle précise la taille de la mémoire tampon utilisée pour stocker l'information tracée. La taille de mémoire tampon peut être définie en octets ou en mégaoctets. Pour l'exprimer en mégaoctets, ajoutez M" ou "m" à la suite de la valeur. Cette taille doit être égale à deux mégaoctets mis à une puissance quelconque. Si vous indiquez une taille ne respectant pas cette obligation, la taille de mémoire tampon est automatiquement arrondie à la puissance la plus proche.

Si la mémoire tampon est trop petite, vous risquez de perdre des informations. Par défaut, seules les informations de trace les plus récentes sont conservées en cas de saturation du tampon. Si la mémoire tampon est trop grande, vous risquez d'avoir des difficultés à envoyer le fichier à l'équipe de support logiciel d'IBM.

Si vous tracez une opération relativement courte (par exemple une connexion de base de données) une taille d'environ 8 Mo suffit en général.

```
C:\> db2trc on -l 8M
Trace is turned on
```

Cependant, si vous tracez une opération plus importante ou si de nombreuses tâches sont effectuées simultanément, une mémoire tampon de trace plus grande peut être nécessaire.

Sur la plupart des plateformes, le traçage peut être activé à tout moment et fonctionne comme indiqué précédemment. Cependant, vous devez avoir connaissance des situations particulières suivantes :

1. Sur les systèmes à partition de base de données multiples, vous devez exécuter une trace pour chaque partition de base de données physique (et non logique).
2. Sous les plateformes HP-UX, Linux et Solaris, si la trace est désactivée une fois l'instance démarrée, une très petite mémoire tampon est utilisée lors du démarrage suivant de la trace, quelle que soit la taille indiquée. Par exemple, hier, vous avez activé la trace à l'aide de la commande `db2trc on -l 8m`, puis avez recueilli la trace et l'avez désactivée (`db2trc off`). Aujourd'hui, vous souhaitez exécuter une trace en définissant la mémoire tampon sur 32 mégaoctets (`db2trc on -l 32m`) sans arrêter l'instance puis redémarrer. Vous vous apercevrez que dans ce cas, la trace ne génère qu'un tampon de petite taille. Pour exécuter correctement une trace sur ces plateformes, activez la trace avant de démarrer l'instance en définissant la taille de tampon souhaitée, puis «videz» ensuite le tampon si nécessaire.

Vidage d'un fichier de trace DB2

Lorsque la fonction de trace a été activée à l'aide de l'option ON, tous les travaux ultérieurs effectués par l'instance sont tracés.

Lorsque la trace est active, l'option `clr` vous permet de vider la mémoire tampon de trace. Toutes les informations existantes de la mémoire tampon de trace sont supprimées.

```
C:\>db2trc clr
Trace has been cleared
```

Lorsque l'opération tracée se termine, utilisez l'option `dmp` suivie du nom de fichier de trace pour vider le contenu de la mémoire tampon sur le disque. Par exemple :

```
C:\>db2trc dmp trace.dmp
Trace has been dumped to file
```

La fonction de trace continue de s'exécuter une fois la mémoire tampon de trace vidée sur le disque. Pour désactiver la fonction de trace, utilisez l'option OFF :

```
C:\>db2trc off
Trace is turned off
```

Formatage d'un fichier de trace DB2

Le fichier de vidage créé par la commande `db2trc dmp` a un format binaire non lisible. Pour vérifier qu'un fichier de trace peut être lu, formatez le fichier de trace binaire de façon à ce qu'il affiche le contrôle de flux, puis envoyez la sortie formatée à une unité null.

L'exemple suivant indique la commande à lancer pour effectuer cette tâche :

```
db2trc flw example.trc nul
```

où `example.trc` est un fichier binaire généré avec l'option `dmp`.

La sortie de cette commande vous signale explicitement tout problème de lecture du fichier et précise si la trace a été ou non encapsulée.

A ce stade, le fichier de vidage peut être envoyé au support logiciel IBM qui le formatera ensuite en fonction de votre niveau de service DB2. Il peut cependant parfois vous être demandé de formater le fichier de vidage au format ASCII avant de l'envoyer. Pour ce faire, utilisez les options `flw` et `fmt`. Vous devez indiquer le nom du fichier de vidage binaire ainsi que le nom du fichier ASCII à créer :

```
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
```

Si cette sortie indique que "Trace wrapped" a la valeur "YES", cela signifie que la mémoire tampon de trace n'était pas assez grande pour contenir toutes les informations collectées pendant la période de trace. Une trace encapsulée peut convenir selon la situation. Si vous êtes intéressé par les informations les plus récentes (c'est l'information par défaut qui est conservée, sauf si l'option `-i` est indiquée), le contenu du fichier de trace est alors suffisant. Toutefois, si vous souhaitez connaître les événements intervenus au tout début de la période de traçage ou la totalité des événements, il est conseillé de relancer l'opération en choisissant une taille de mémoire tampon plus grande.

Plusieurs options sont disponibles lorsque vous formatez un fichier binaire en fichier texte lisible. Par exemple, vous pouvez utiliser la commande `db2trc fmt -xml trace.dmp trace.fmt` pour convertir les données binaires et envoyer le résultat dans un format analysable XML. Les autres options s'affichent dans la description détaillée de la commande de trace (`db2trc`).

Vous devez également savoir que sur les systèmes d'exploitation Linux et UNIX, DB2 vide automatiquement la mémoire tampon de trace sur le disque lorsqu'il arrête l'instance suite à une erreur grave. Ainsi, si la trace est activée lorsqu'une instance termine de façon anormale, un fichier est créé dans le répertoire de diagnostic sous le nom de db2trdmp.###, où ### est le numéro de la partition de la base de données. Cela ne se produit pas sur les plateformes Windows. Vous devez vider la trace manuellement dans ces cas.

Pour récapituler, voici un exemple la séquence courante de commandes db2trc :

```
db2trc on -l 8M
db2trc clr
<Execute problem recreation commands>
db2trc dump db2trc.dmp
db2trc off
db2trc flw db2trc.dmp <filename>.flw
db2trc fmt db2trc.dmp <filename>.fmt
db2trc fmt -c db2trc.dmp <filename>.fmtc
```

Chapitre 24. Fichiers de trace DRDA

Avant d'analyser les traces DRDA, vous devez avoir clairement assimilé la notion que DRDA est une norme ouverte de définition des structures de données et de communications. Par exemple, DRDA comprend un ensemble de règles indiquant comment les données doivent être organisées en vue de leur transmission et comment doit se dérouler la communication de ces informations.

Ces règles sont définies dans les manuels de référence suivants :

- 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

Des versions PDF de ces manuels sont disponibles sur le site www.opengroup.org.

L'utilitaire **db2drdat** enregistre les données échangées entre un demandeur d'application DRDA Application Requestor (AR) et un serveur d'application DB2 DRDA Application Server (AS) (par exemple entre DB2 Connect et un serveur de base de données hôte ou Power Systems Servers).

Utilitaire de trace

L'utilitaire **db2drdat** enregistre les données échangées entre DB2 Connect (pour le compte du client IBM Data Server) et le serveur de base de données grand système IBM.

En tant qu'administrateur de base de données (ou développeur de l'application), vous pouvez trouver utile de connaître le fonctionnement de ce flot de données car cela peut vous aider à déterminer l'origine d'un incident déterminé. Supposons par exemple la situation suivante : vous avez émis une instruction de base de données **CONNECT TO** pour un serveur de base de données grand système IBM mais la commande a échoué en renvoyant un code retour d'échec. Si vous savez exactement quelles informations ont été acheminées vers le système de gestion du serveur de base de données grand système IBM, vous pourrez peut-être déterminer l'origine de l'échec même si les informations du code retour sont d'ordre général. De nombreux incidents sont provoqués par de simples erreurs de l'utilisateur.

La sortie de **db2drdat** répertorie les flots de données échangés entre le poste de travail DB2 Connect et le système de gestion du serveur de base de données grand système IBM. Les données envoyées au serveur de base de données grand système IBM s'intitulent **SEND BUFFER** et celles reçues du serveur de base de données grand système IBM s'intitulent **RECEIVE BUFFER**.

Si une mémoire tampon de réception contient des informations SQLCA, elle sera suivie d'une interprétation formatée de ces données et de la SQLCA qualifiée. La zone SQLCODE d'une SQLCA représente la valeur *unmapped* (non mappée) renvoyée par le serveur de base de données grand système IBM. Les mémoires tampon d'envoi et de réception sont classées par ordre d'ancienneté, de la plus ancienne à plus récente, dans le fichier. Chaque mémoire tampon possède :

- L'ID processus
- SEND BUFFER, RECEIVE BUFFER ou un intitulé SQLCA. La première commande DDM ou le premier objet DDM d'une mémoire tampon qualifié(e) en tant que DDS TYPE.

Les autres données présentes dans les mémoires tampon d'envoi et de réception sont réparties en cinq colonnes qui se composent :

- d'un comptage d'octets.
- des colonnes 2 et 3 qui représentent le flot de données DRDA entre les deux systèmes, en code ASCII ou EBCDIC.
- d'une représentation en code ASCII des colonnes 2 et 3.
- d'une représentation en code EBCDIC des colonnes 2 et 3.

Sortie de trace

L'utilitaire db2drdat inscrit les informations suivantes dans le *fichier de trace* :

- -r
 - Type de réponse/d'objet DRDA
 - Mémoire tampon de réception
- -s
 - Type de requête DRDA
 - Mémoire tampon d'envoi
- -c
 - SQLCA
- Informations relatives à l'erreur TCP/IP
 - Code retour de la fonction de réception
 - Gravité
 - Protocole utilisé
 - API utilisée
 - Fonction
 - Numéro d'erreur.

Remarque :

1. Une valeur égale à zéro pour le code de sortie indique que la commande a été exécutée avec succès et une valeur différente de zéro indique que ce n'est pas le cas.
2. Les champs renvoyés varient en fonction de l'API utilisée.
3. Les champs renvoyés varient en fonction de la plateforme sur laquelle DB2 Connect fonctionne, même lorsque l'API est identique.
4. Si la commande db2drdat envoie les résultats dans un fichier qui existe déjà, l'ancien fichier sera effacé sauf si les droits d'accès au fichier ne le permettent pas.

Analyse du fichier de sortie de trace

Les informations suivantes sont enregistrées dans un fichier de trace db2drdat :

- L'ID de processus (PID) de l'application client
- Le RDB_NAME enregistré dans le répertoire des services de connexion à la base de données
- Le(s) CCSID DB2 Connect
- Le(s) CCSID du serveur de base de données grand système IBM
- Le système de gestion du serveur de base de données grand système IBM avec lequel communique le système DB2 Connect.

La première mémoire tampon contient les commandes EXCSAT (Exchange Server Attributes) et ACCRDB (Access RDB) envoyées au système de gestion du serveur de base de données grand système IBM. Elle envoie ces commandes suite à une commande de base de données CONNECT T0. La mémoire tampon suivante contient la réponse reçue par DB2 Connect du système de gestion du serveur de base de données grand système IBM. Elle contient les données de réponse Exchange Server Attributes (EXCSATRD) et un message de réponse Access RDB (ACCRDBRM).

EXCSAT

La commande EXCSAT contient le nom du poste de travail du client spécifié par l'objet Nom de serveur (SRVNAM) qui est un point de code X'116D', conformément à la spécification DDM. La commande EXCSAT se trouve dans la première mémoire tampon. Dans la commande EXCSAT, les valeurs X'9481A292' (codées dans le CCSID 500) sont traduites en *masque* lorsque X'116D' est supprimé.

La commande EXCSAT contient également l'objet EXTNAM (External Name), souvent placé dans les données de diagnostic du système de gestion de la base de données grand système IBM. Elle se compose d'une application de 20 octets suivie d'un ID processus de 8 octets (ou d'un ID processus de 4 octets et d'un ID d'unité d'exécution de 4 octets). Elle est représentée par un point de code X'115E', et, dans cet exemple, sa valeur est db2bp intercalée de blancs et suivie de 000C50CC. Sur un client IBM Data Server Linux ou UNIX, cette valeur peut être corrélée avec la commande ps, qui renvoie des informations sur le statut des processus actifs dans une sortie standard.

ACCRDB

La commande ACCRDB contient le RDB_NAME dans l'objet RDBNAM, qui est le point de code X'2110'. La commande ACCRDB vient après la commande EXCSAT dans la première mémoire tampon. Dans la commande ACCRDB, les valeurs X'E2E3D3C5C3F1' sont traduites en STLEC1 lorsque X'2110' est supprimé. Cela correspond au champ du nom de la base de données cible dans le répertoire DCS.

L'identifiant comptable possède le code de point X'2104'.

Le jeu de codes configuré pour le poste de travail DB2 Connect s'affiche en localisant l'objet CCSID CCSIDSBC (CCSID pour les caractères mono-octet) avec le code de point X'119C' de la commande ACCRDB. Dans cet exemple, le CCSIDSBC est X'0333', représentant la valeur 819.

Les objets supplémentaires CCSIDDBC (CCSID pour les caractères codés sur deux octets) et CCSIDMBC (CCSID pour les caractères mixtes) avec les codes de point respectifs X'119D' et X'119E' sont également présents dans la commande ACCRDB. Dans cet exemple, le CCSIDDBC est X'04B0', représentant la valeur 1200, et le CCSIDMBC est X'0333', représentant la valeur 819.

EXCSATRD et ACCRDBRM

Les valeurs CCSID sont également renvoyées à partir du serveur de base de données grand système IBM dans le message de réponse ACCRDBRM (Access RDB Reply Message) de la seconde mémoire tampon. Cette mémoire tampon contient les données EXCSATRD suivies du message ACCRDBRM. L'exemple de fichier de sortie contient deux valeurs CCSID pour le système du serveur de base de données grand système IBM. Ces valeurs sont de 1208 (pour les caractères mono-octet et mixtes) et de 1200 (pour les caractères codés sur deux octets).

Si DB2 Connect ne reconnaît pas la page de codes renvoyées par le serveur de base de données grand système IBM, SQLCODE -332 sera renvoyé à l'utilisateur avec les pages de code source et cible. Si le serveur de base de données grand système IBM ne reconnaît pas le jeu de codes renvoyé par DB2 Connect, il renverra la valeur VALNSPRM (Valeur de paramètre non prise en charge, avec le point de code DDM X'1252'), qui sera converti en SQLCODE -332 pour l'utilisateur.

Le message ACCRDBRM contient également le paramètre PRDID (identifiant spécifique au produit, avec le point de code X'112E'). La valeur est X'C4E2D5F0F8F0F1F5' qui correspond à DSN08015 dans EBCDIC. Les normes stipulent que DSN équivaut à DB2 for z/OS. Le numéro de version est également indiqué. ARI correspond à DB2 Server for VSE & VM, SQL à une base de données DB2 ou DB2 Connect, et QSQ à DB2 for IBM i.

Modèles de fichiers de sortie de trace

Les exemples ci-dessous illustrent divers flots de données DRDA échangés entre des postes de travail DB2 Connect et un hôte ou un serveur de base de données System i. Du point de vue de l'utilisateur, une commande de base de données CONNECT TO a été exécutée à l'aide de l'interpréteur de commandes (CLP).

La figure 13, à la page 165 utilise DB2 Connect Enterprise Edition version 9.1 et DB2 for z/OS version 8 via une connexion 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 F261C1C9E7F6F400G....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 F20014116DE2E3D3G.....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

Figure 13. Exemple de sortie de trace (connexion 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	4040404040000C111
0040	2EE2D8D3F0F9F0F0	F000D002FD8E3C4/...SQL0900....QTD
0050	E2D8D3C1E2C30016	00350006119C03335.....3
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	.../.....
0020	2EC4E2D5F0F8F0F1	F5001600350006115....
0030	9C04B80006119E04	B80006119D04B000
0040	0C11A0D5C5E6E3D6	D540400006212524@...!%\$
0050	34001E244E000624	4C00010014244D00	4..\$N..\$L...\$M.
0060	06244FFFFF000A11	E8091E768301BE00	.\$0.....v....
0070	2221030000000005	68B3B8C7F9F1C1F0	"!.....h.....
0080	C4F3C1D7C1F8F840	4040400603022106@@@...!
0090	46000A11E8091E76	831389	F.....v....

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

Figure 14. Exemple de sortie de trace (connexion TCP/IP)- suite

```

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

```

Figure 15. Exemple de sortie de trace (connexion TCP/IP)- suite

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	00000000000572020 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\$..	..}.....

Figure 16. Exemple de sortie de trace (connexion TCP/IP)- suite

Informations de mémoire tampon postérieures pour les traces DRDA

Vous pouvez analyser des mémoires tampon de réception et d'envoi ultérieures pour obtenir des informations supplémentaires. La requête suivante contient une validation. La commande commit indique au système de gestion du serveur de base de données grand système IBM de valider l'unité d'oeuvre actuelle. La quatrième mémoire tampon est reçue du système de gestion du serveur de base de données grand système IBM suite à une validation ou à une annulation de l'opération. Elle contient le message de réponse de fin d'unité d'oeuvre (ENDUOWRM) qui indique que l'unité d'oeuvre courante est achevée.

Dans cet exemple, l'entrée de trace 12 contient une SQLCA de valeur nulle, indiquée par le point de code X'2408' suivi de X'FF'. Une SQLCA de valeur nulle (X'2408FF') indique le succès de l'opération (SQLCODE 0).

La figure 13, à la page 165 illustre un exemple d'une mémoire tampon de réception contenant une erreur SQLCA au niveau de trace 16.

Partie 6. Messages

Chapitre 25. Incidents DB2 Connect courants

Cette rubrique répertorie les symptômes les plus courants des incidents rencontrés lorsque vous utilisez DB2 Connect. Quel que soit le cas de figure, vous obtenez :

- Une combinaison d'un numéro de message et d'un code retour (ou un code retour spécifique au protocole) associée à ce message. Chaque combinaison de message et de code retour possède un en-tête spécifique. Ces en-têtes sont classés par numéro de message, puis par code retour.
- Un symptôme, généralement sous la forme d'une liste d'exemples de messages.
- Une proposition de solution, indiquant l'origine probable de l'erreur. Dans certains cas de figure, plusieurs solutions peuvent être proposées.

SQL0965 ou SQL0969

Symptôme

Les messages SQL0965 et SQL0969 peuvent être émis avec plusieurs codes retour différents provenant de DB2 for IBM i, DB2 for z/OS, et DB2 Server for VM and VSE.

Lorsque vous rencontrez l'un de ces messages, vérifiez le code SQL d'origine dans la documentation relative au serveur de base de données qui émet le message.

Solution

Le code SQL reçu de la base de données grand système IBM ne peut pas être converti. Corrigez l'incident en vous basant sur le code d'erreur, puis soumettez à nouveau la commande défectueuse.

SQL5043N

Symptôme

Le démarrage du support d'un ou de plusieurs protocoles a échoué. Toutefois, la fonction du gestionnaire de bases de données a pu être démarrée.

Il se peut que le protocole TCP/IP ne soit pas démarré sur le serveur DB2 Connect. Une connexion client réussie peut avoir eu lieu précédemment.

Si `diaglevel = 4`, les fichiers journaux `db2diag` peuvent contenir une entrée similaire à celle ci-dessous :

```
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.
```

Solution

Cet avertissement est un symptôme signalant que DB2 Connect, qui agit en tant que serveur pour les clients distants, rencontre des difficultés à gérer un ou plusieurs protocoles de communication client. Il peut s'agir du protocole TCP/IP ou d'autres protocoles. En règle générale, ce message indique que l'un des protocoles de communication définis dans DB2 Connect n'est pas configuré correctement.

L'origine de ce message peut être une mauvaise définition de la variable de profil DB2COMM ou l'absence de définition de cette variable. En règle générale, l'incident est le résultat d'une incohérence entre la variable DB2COMM et les noms définis dans la configuration du gestionnaire de base de données (par exemple, svcename ou nname).

Un scénario possible consiste à avoir réussi à établir une connexion précédente, puis à obtenir le message d'erreur SQL5043 alors qu'aucune configuration n'a été modifiée. Ce scénario peut se produire lorsque vous utilisez le protocole TCP/IP et que le système distant met fin à la connexion de façon anormale pour une raison quelconque. Si ce scénario se produit, une connexion peut toujours sembler exister sur le client et il peut devenir impossible de restaurer la connexion sans exécution supplémentaire des commandes suivantes.

Il est très probable que l'un des clients qui se connectent au serveur DB2 Connect possède toujours un descripteur sur le port TCP/IP. Sur chaque machine client connectée au serveur DB2 Connect, saisissez les commandes suivantes :

```
db2 terminate
db2stop
```

SQL30020

Symptôme

SQL30020N L'exécution a échoué en raison d'une erreur dans le protocole de répartition qui empêche l'exécution réussie des commandes et instructions SQL suivantes.

Solutions

Si cette erreur se produit, vous devez contacter la maintenance. Exécutez la commande db2support avant de contacter le service de maintenance.

SQL30060

Symptôme

SQL30060N "<ID-autorisation>" ne dispose pas du privilège permettant d'exécuter l'opération <opération>".

Solution

Lors de la connexion à DB2 for z/OS, les tables de bases de données de communications (CDB) n'ont pas été mises à jour correctement.

SQL30061

Symptôme

Connexion à un emplacement erroné de serveur de base de données grand système IBM - Aucune base de données cible n'a pu être trouvée.

Solution

Il se peut qu'un mauvais nom de base de données du serveur soit spécifié dans l'entrée de répertoire DCS. Si tel est le cas, le code SQLCODE -30061 est renvoyé à l'application.

Vérifiez le noeud DB2, la base de données et les entrées de répertoire DCS. Le champ correspondant au nom de la base de données cible dans l'entrée du répertoire DCS doit correspondre au nom de la base de données basée sur la plateforme. Par exemple, pour une base de données DB2 for z/OS, le nom à utiliser devrait être identique au nom utilisé dans la zone LOCATION=*locname*", du fichier d'amorçage, qui est également indiqué

dans le message DSNL004I (LOCATION=*location*) lors du lancement de l'utilitaire DDF (Distributed Data Facility).

Les commandes valides pour un noeud TCP/IP sont :

```
db2
catalog tcpip node <nom_noeud> remote
<nom_ou_adresse_hôte>
server <no_port_ou_nom_service>
db2 catalog dcs database <nom_local> as
<nom_réel_bd>
db2 catalog database <nom_local> as <alias> at
<nom_noeud noeud>
authentication server
```

Pour vous connecter à la base de données, exécutez la commande suivante :

```
db2
connect to <alias> user <nom_utilisateur> using
<mot de passe>
```

SQL30081N avec le code retour 79

Symptôme

```
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
```

Solution(s)

Cette erreur peut se produire lorsqu'un client distant ne parvient pas à se connecter au serveur DB2 Connect. Aucune base de données cible n'a pu être trouvée du serveur DB2 Connect à un serveur de base de données grand système IBM.

1. La variable de profil DB2COMM peut être définie de façon incorrecte dans le serveur DB2 Connect. Vérifiez la définition de la variable. Par exemple, la commande db2set db2comm=tcpip doit apparaître dans sql1lib/db2profile lorsque vous exécutez DB2 Enterprise Server Edition sous AIX.
2. Il peut exister une non-concordance entre les spécifications du nom de service ou du numéro de port TCP/IP sur le client IBM Data Server et le serveur DB2 Connect. Vérifiez les fichiers des services TCP/IP sur les deux machines.
3. Vérifiez que DB2 est démarré sur le serveur DB2 Connect. Définissez le diaglevel de la configuration du gestionnaire de base de données sur 4, à l'aide de la commande suivante :

```
db2 update dbm cfg using diaglevel 4
```

Après avoir arrêté puis redémarré DB2, vérifiez dans les fichiers journaux db2diag que des communications TCP/IP DB2 ont été initiées. Vous apercevrez une sortie similaire à celle-ci :

```
2001-02-03-12.41.04.861119 Instance:svtdbm2 Node:00
PID:86496(db2sysc) Appid:none
common communication sqlcctcp_start_listen Probe:80
DIA3000I "TCP/IP" protocol support was successfully started.
```

SQL30081N avec le code d'erreur spécifique au protocole 10032

Symptôme

```
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
```

Solution

Vous pouvez recevoir ce message d'erreur lorsque vous tentez de vous déconnecter d'une machine sur laquelle des communications TCP/IP ont déjà échoué. Corrigez l'incident à l'aide du sous-système TCP/IP.

Sur la plupart des machines, un simple redémarrage du protocole TCP/IP suffit à résoudre l'incident. Un recyclage éventuel de la totalité de la machine peut être nécessaire.

SQL30082 RC=24 lors de CONNECT

Symptôme

SQLCODE -30082 Le nom d'utilisateur ou le mot de passe est incorrect.

Solution

Vérifiez que le bon mot de passe est fourni dans l'instruction CONNECT en cas de besoin. Mot de passe indisponible pour l'envoi à la base de données du serveur cible. Un mot de passe doit être envoyé du client IBM Data Server vers la base de données du serveur cible. Sur certaines plateformes, par exemple AIX, le mot de passe peut uniquement être obtenu via l'instruction CONNECT.

Partie 7. Annexes

Annexe A. Présentation des informations techniques DB2

Les informations techniques DB2 sont disponibles via les méthodes et les outils suivants :

- Centre de documentation DB2
 - Rubriques (tâches, concepts et référence)
 - Aide sur les outils DB2
 - Exemples de programmes
 - Tutoriels
- Manuels DB2
 - Fichiers PDF (téléchargeables)
 - Fichiers PDF (se trouvant sur le DVD des documents PDF DB2)
 - Manuels imprimés
- Aide sur les lignes de commande
 - Aide sur la commande
 - Aide sur le message

Remarque : Les rubriques du centre de documentation DB2 sont mises à jour plus régulièrement que les fichiers PDF ou les manuels en version papier. Pour avoir accès aux informations les plus récentes, installez les mises à jour de la documentation dès qu'elles sont disponibles ou consultez le centre de documentation DB2 sur le site ibm.com.

Vous pouvez accéder à des informations techniques DB2 supplémentaires, telles que les notes techniques, les livres blancs et les documents IBM Redbooks disponibles en ligne sur le site ibm.com. Accédez au site de la bibliothèque des logiciels de gestion des informations DB2 à l'adresse <http://www.ibm.com/software/data/sw-library/>.

Commentaires sur la documentation

Nous accordons une grande importance à vos commentaires sur la documentation DB2. Si vous avez des suggestions permettant d'améliorer la documentation DB2, envoyez un message électronique à db2docs@ca.ibm.com. L'équipe de documentation DB2 lit tous les commentaires mais ne peut pas vous répondre directement. Indiquez des exemples précis, lorsque cela est possible, afin que nous puissions mieux comprendre vos préoccupations. Si vous avez des commentaires sur une rubrique ou un fichier d'aide spécifique, indiquez le titre de la rubrique et l'URL.

N'utilisez pas cette adresse électronique pour contacter le service clients DB2. Si vous rencontrez un problème technique DB2 non résolu par la documentation, contactez le service de maintenance IBM.

Bibliothèque technique DB2 au format PDF ou en version papier

Le tableau suivant décrit la bibliothèque DB2 disponible dans le centre de publications IBM à l'adresse suivante www.ibm.com/shop/publications/order. Vous pouvez télécharger la version anglaise ainsi que les versions traduites des manuels DB2 version 9.7 au format PDF à l'adresse suivante : www.ibm.com/support/docview.wss?rs=71&uid=swg2700947.

Ces tableaux identifient les documents disponibles au format papier, mais il se peut que ces derniers ne soient pas disponibles dans votre pays ou votre région.

Le numéro de référence d'un document est incrémenté à chaque mise à jour de ce document. Prenez soin de consulter la version la plus récente de ces manuels, tel qu'indiqué ci-dessous.

Remarque : Le centre de documentation DB2 est mis à jour plus fréquemment que les fichiers PDF ou les manuels en version imprimée.

Tableau 20. Informations techniques sur DB2

Nom	Référence	Disponible au format papier	Dernière mise à jour
<i>Administrative API Reference</i>	SC27-2435-00	Oui	Août 2009
<i>Administrative Routines and Views</i>	SC27-2436-00	Non	Août 2009
<i>Call Level Interface Guide and Reference, Volume 1</i>	SC27-2437-00	Oui	Août 2009
<i>Call Level Interface Guide and Reference, Volume 2</i>	SC27-2438-00	Oui	Août 2009
<i>Command Reference</i>	SC27-2439-00	Oui	Août 2009
<i>Data Movement Utilities Guide and Reference</i>	SC27-2440-00	Oui	Août 2009
<i>Data Recovery and High Availability Guide and Reference</i>	SC27-2441-00	Oui	Août 2009
<i>Database Administration Concepts and Configuration Reference</i>	SC27-2442-00	Oui	Août 2009
<i>Database Monitoring Guide and Reference</i>	SC27-2458-00	Oui	Août 2009
<i>Database Security Guide</i>	SC27-2443-00	Oui	Août 2009
<i>DB2 Text Search Guide</i>	SC27-2459-00	Oui	Août 2009
<i>Developing ADO.NET and OLE DB Applications</i>	SC27-2444-00	Oui	Août 2009
<i>Developing Embedded SQL Applications</i>	SC27-2445-00	Oui	Août 2009
<i>Developing Java Applications</i>	SC27-2446-00	Oui	Août 2009

Tableau 20. Informations techniques sur DB2 (suite)

Nom	Référence	Disponible au format papier	Dernière mise à jour
<i>Developing Perl, PHP, Python, and Ruby on Rails Applications</i>	SC27-2447-00	Non	Août 2009
<i>Developing User-defined Routines (SQL and External)</i>	SC27-2448-00	Oui	Août 2009
<i>Getting Started with Database Application Development</i>	GI11-9410-00	Oui	Août 2009
<i>Guide d'initiation à l'installation et à l'administration de DB2 sous Linux et Windows</i>	GI11-7343-00	Oui	Août 2009
<i>Globalization Guide</i>	SC27-2449-00	Oui	Août 2009
<i>Installation de serveurs DB2</i>	GC11-6570-00	Oui	Août 2009
<i>Installation de clients IBM Data Server</i>	GC11-6571-00	Non	Août 2009
<i>Guide des messages, volume 1</i>	SC11-6576-00	Non	Août 2009
<i>Guide des messages, volume 2</i>	SC11-6577-00	Non	Août 2009
<i>Net Search Extender - Guide d'administration et d'utilisation</i>	SC11-6579-00	Non	Août 2009
<i>Partitioning and Clustering Guide</i>	SC27-2453-00	Oui	Août 2009
<i>pureXML Guide</i>	SC27-2465-00	Oui	Août 2009
<i>Query Patroller - Guide d'administration et d'utilisation</i>	SC11-6580-00	Non	Août 2009
<i>Spatial Extender and Geodetic Data Management Feature User's Guide and Reference</i>	SC27-2468-00	Non	Août 2009
<i>SQL Procedural Languages: Application Enablement and Support</i>	SC27-2470-00	Oui	Août 2009
<i>SQL Reference, Volume 1</i>	SC27-2456-00	Oui	Août 2009
<i>SQL Reference, Volume 2</i>	SC27-2457-00	Oui	Août 2009
<i>Troubleshooting and Tuning Database Performance</i>	SC27-2461-00	Oui	Août 2009
<i>Mise à niveau vers DB2 version 9.7</i>	SC11-6569-00	Oui	Août 2009
<i>Tutoriel Visual Explain</i>	SC11-6578-00	Non	Août 2009

Tableau 20. Informations techniques sur DB2 (suite)

Nom	Référence	Disponible au format papier	Dernière mise à jour
<i>Nouveautés de DB2 version 9.7</i>	SC11-6575-00	Oui	Août 2009
<i>Workload Manager Guide and Reference</i>	SC27-2464-00	Oui	Août 2009
<i>XQuery Reference</i>	SC27-2466-00	Non	Août 2009

Tableau 21. Informations techniques spécifiques de DB2 Connect

Nom	Référence	Disponible au format papier	Dernière mise à jour
<i>Installation et configuration de DB2 Connect Personal Edition</i>	SC11-6573-00	Oui	Août 2009
<i>Installation et configuration de serveurs DB2 Connect</i>	SC11-6574-00	Oui	Août 2009
<i>DB2 Connect - Guide d'utilisation</i>	SC11-6572-00	Oui	Août 2009

Tableau 22. Informations techniques sur Information Integration

Nom	Référence	Disponible au format papier	Dernière mise à jour
<i>Information Integration: Administration Guide for Federated Systems</i>	SC19-1020-02	Oui	Août 2009
<i>Information Integration : Référence du programme ASNCLP pour la réplication et la publication</i>	SC11-2663-03	Oui	Août 2009
<i>Information Integration: Configuration Guide for Federated Data Sources</i>	SC19-1034-02	Non	Août 2009
<i>Information Integration : Guide de référence de la réplication SQL</i>	SC11-6545-00	Oui	Août 2009
<i>Information Integration : Introduction à la réplication et à la publication d'événement</i>	GC11-6528-00	Oui	Août 2009

Commande de manuels imprimés DB2

Si vous avez besoin de manuels imprimés DB2, vous pouvez les acheter en ligne dans un grand nombre de pays ou de régions. Vous pouvez toujours commander des manuels DB2 imprimés auprès de votre représentant IBM. Gardez à l'esprit que certains manuels au format électronique sur le DVD de la *documentation PDF DB2* ne sont pas disponibles au format imprimé. Par exemple, aucun des volumes *Guide des messages DB2* n'est disponible sous forme de documentation imprimée.

Les versions imprimées de nombreux documents DB2 disponibles sur le DVD de la documentation PDF DB2 sont en vente auprès d'IBM. Suivant votre lieu de résidence, vous pouvez commander des documents en ligne à partir de l'IBM Publications Center. Si les commandes en ligne ne sont pas disponibles dans votre pays ou votre région, vous pouvez toujours commander les documents DB2 imprimés auprès de votre représentant IBM. Notez que les documents du DVD de documentation PDF DB2 ne sont pas tous disponibles au format papier.

Remarque : La documentation complète de DB2 la plus récente est à votre disposition dans le centre de documentation DB2 à l'adresse suivante : <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7>.

Pour commander des documents DB2 imprimés, procédez comme suit :

- Pour savoir s'il est possible de commander des documents imprimés DB2 dans votre pays ou votre région, consultez l'IBM Publications Center à l'adresse suivante <http://www.ibm.com/shop/publications/order>. Vous devez sélectionner un pays, une région ou une langue pour accéder aux informations de commande des publications et suivre les instructions permettant de passer une commande là où vous résidez.
- Pour commander des documents imprimés DB2 auprès de votre représentant IBM, procédez comme suit :
 1. Recherchez les coordonnées de votre représentant local sur l'un des sites Web suivants :
 - L'annuaire IBM international des contacts à l'adresse suivante : www.ibm.com/planetwide
 - Le site Web des publications IBM à l'adresse suivante : <http://www.ibm.com/shop/publications/order>. Vous devez sélectionner votre pays, région ou langue pour accéder à la page d'accueil des publications appropriée. Dans cette page, suivez le lien "About this site".
 2. Si vous appelez, précisez que vous souhaitez commander une publication DB2.
 3. Indiquez à votre représentant les titres et les numéros de référence des manuels que vous souhaitez commander. Pour plus de détails, voir «Bibliothèque technique DB2 au format PDF ou en version papier», à la page 180.

Affichage de l'aide sur les codes d'état SQL à partir de l'interpréteur de commandes

Les produits de la famille DB2 renvoient une valeur SQLSTATE pour les conditions qui peuvent être le résultat d'une instruction SQL. L'aide sur les états SQL (SQLSTATE) donne la signification des états SQL et des codes de classe de ces états.

Pour lancer l'aide sur les états SQL, ouvrez l'interpréteur de commandes et tapez :

`? sqlstate` ou `? code-classe`

où `sqlstate` correspond à un code d'état SQL correct composé de cinq chiffres et `code-classe` aux deux premiers chiffres du code d'état SQL.

Par exemple, `? 08003` permet d'afficher l'aide sur l'état SQL 08003 et `? 08` permet de visualiser l'aide sur le code de classe 08.

Accès aux différentes versions du centre de documentation DB2

Pour les rubriques DB2 version 9.7, l'URL du centre de documentation DB2 est <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/>

Pour les rubriques DB2 version 9.5, l'URL du centre de documentation DB2 est <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>

Pour les rubriques DB2 version 9, l'URL du centre de documentation DB2 est <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>

Pour les rubriques DB2 version 8, accédez à l'URL du centre de documentation de la version 8 à l'adresse suivante :

<http://publib.boulder.ibm.com/infocenter/db2luw/v8/>

Affichage des rubriques dans votre langue préférée dans le centre de documentation DB2

Le centre de documentation DB2 affiche les rubriques dans la langue définie dans les préférences de votre navigateur. Si la rubrique n'est pas disponible dans cette langue, le centre de documentation DB2 affiche la version anglaise.

- Pour afficher les rubriques dans votre langue préférée dans le navigateur Web Internet Explorer, procédez comme suit :
 1. Dans Internet Explorer, sélectionnez **Outils** —> **Options Internet** —> **Langues**. La fenêtre Langues s'ouvre.
 2. Vérifiez que votre langue préférée est indiquée dans la première entrée de la liste de langues.
 - Pour ajouter une langue à la liste, cliquez sur le bouton **Ajouter...**
 3. Videz la mémoire cache du navigateur puis régénérez la page afin d'afficher le centre de documentation DB2 dans la langue choisie.
- Pour afficher les rubriques dans la langue de votre choix dans un navigateur Firefox ou Mozilla :
 1. Sélectionnez le bouton dans la section **Langues** de la boîte de dialogue **Outils** —> **Options** —> **Paramètres avancés**. Le panneau Langues est affiché dans la fenêtre Préférences.
 2. Vérifiez que votre langue préférée est indiquée dans la première entrée de la liste de langues.
 - Pour ajouter une nouvelle langue à la liste, cliquez sur le bouton **Ajouter...** afin de la sélectionner dans la fenêtre Ajouter des langues.
 - Pour faire passer une langue en haut de la liste, sélectionnez-la et cliquez sur le bouton **Monter** jusqu'à ce qu'elle apparaisse en premier.
 3. Videz la mémoire cache du navigateur puis régénérez la page afin d'afficher le centre de documentation DB2 dans la langue choisie.

Pour certaines combinaisons de navigateur et de système d'exploitation, vous devez également modifier les paramètres régionaux de votre système d'exploitation pour spécifier l'environnement local et la langue de votre choix.

Mise à jour du centre de documentation DB2 installé sur votre ordinateur ou sur votre serveur intranet

Un centre de documentation DB2 local doit être mis à jour régulièrement.

Avant de commencer

Un centre de documentation DB2 version 9.7 doit déjà être installé. Pour plus d'informations, voir la rubrique «Installation du centre de documentation DB2 avec l'assistant d'installation DB2» dans *Installation de serveurs DB2*. Toutes les conditions prérequis et les restrictions s'appliquant au centre de documentation s'appliquent également à sa mise à jour.

A propos de cette tâche

Un centre de documentation DB2 existant peut être mis à jour automatiquement ou manuellement :

- Mises à jour automatiques - mise à jour des fonctions et langues d'un centre de documentation existant. Les mises à jour automatiques offrent l'avantage supplémentaire de ne rendre le centre de documentation indisponible que pendant une durée limitée. De plus, les mises à jour automatiques peuvent être définies de façon à s'exécuter au sein d'autres travaux par lots sur une base régulière.
- Mises à jour manuelles - préférez une mise à jour manuelle lorsque vous souhaitez ajouter des fonctions ou des langues pendant le processus de mise à jour. Par exemple, vous souhaitez ajouter l'allemand à un centre de documentation installé à l'origine avec les seules langues anglaise et française. Dans ce cas, exécutez une mise à jour manuelle pour installer l'allemand tout en mettant à jour les fonctions et langues. Notez cependant que pour une mise à jour manuelle, vous devez arrêter, mettre à jour et redémarrer vous-même le centre de documentation. Le centre de documentation est ainsi indisponible pendant toute la durée du processus de mise à jour.

Procédure

Cette rubrique décrit le processus de la mise à jour automatique. Pour consulter les instructions concernant la mise à jour manuelle, voir la rubrique «Mise à jour manuelle du centre de documentation DB2 installé sur votre ordinateur ou serveur intranet».

Pour mettre à jour automatiquement le centre de documentation DB2 installé sur votre ordinateur ou serveur intranet :

1. Pour les systèmes d'exploitation Linux,
 - a. Accédez au chemin d'installation du centre de documentation. Par défaut, le centre de documentation DB2 se trouve dans le répertoire `/opt/ibm/db2ic/version 9.7`.
 - b. A partir du répertoire d'installation, accédez au répertoire `doc/bin`.
 - c. Exécutez le script `ic-update` :
`ic-update`

2. Pour les systèmes d'exploitation Windows :
 - a. Ouvrez une fenêtre de commande.
 - b. Accédez au chemin d'installation du centre de documentation. Par défaut, le centre de documentation DB2 est installé dans le répertoire <Program Files>\IBM\DB2 Information Center\Version 9.7, où <Program Files> représente l'emplacement du répertoire Program Files.
 - c. A partir du répertoire d'installation, accédez au répertoire doc\bin.
 - d. Exécutez le fichier ic-update.bat :

```
ic-update.bat
```

Résultats

Le centre d'information DB2 redémarre automatiquement. Si des mises à jour ont été trouvées, le centre de documentation affiche les rubriques nouvelles ou mises à jour. Si aucune mise à jour n'a été trouvée, un message est ajouté au journal. Le fichier journal se trouve dans le répertoire doc\eclipse\configuration. Le nom du fichier journal est un nombre généré de façon aléatoire. Par exemple, 1239053440785.log.

Mise à jour manuelle du centre de documentation DB2 installé sur votre ordinateur ou sur votre serveur intranet

Si vous avez installé le centre de documentation DB2 localement, vous pouvez obtenir auprès d'IBM les mises à jour de cette documentation et les installer.

Pour la mise à jour du centre de documentation DB2 installé localement, procédez comme suit :

1. Arrêtez le centre de documentation DB2 sur votre ordinateur et redémarrez le centre de documentation en mode autonome. Son exécution en mode autonome empêche les autres utilisateurs du réseau d'y accéder et vous permet de lui appliquer des mises à jour. La Version poste de travail du Centre de documentation DB2 s'exécute toujours en mode autonome.
2. Vérifiez quelles mises à jour sont disponibles à l'aide de la fonctionnalité de mise à jour. Installez ensuite les mises à jour à l'aide de cette fonctionnalité.

Remarque : Si votre environnement nécessite l'installation des mises à jour du centre de documentation DB2 sur une machine qui n'est pas connectée à Internet, mettez en miroir le site de mise à jour sur un système de fichier local via une machine connectée à Internet sur laquelle est installé le centre de documentation DB2. Si beaucoup d'utilisateurs du réseau doivent installer les mises à jour de documentation, vous pouvez leur faire gagner du temps lors de l'exécution de cette procédure en effectuant une mise en miroir du site localement, puis en créant un proxy pour le site de mise à jour.

Le cas échéant, utilisez la fonction de mise à jour pour vous procurer les modules. Sachez toutefois que cette fonction n'est disponible qu'en mode autonome.

3. Arrêtez le centre de documentation autonome et redémarrez le centre de documentation DB2 sur votre ordinateur.

Remarque : Sous Windows 2008, Windows Vista (et les versions supérieures), les commandes répertoriées ci-après dans cette section doivent être exécutées en tant qu'administrateur. Pour ouvrir une invite de commande ou un outil graphique avec droits d'administrateur complets, cliquez sur le raccourci et sélectionnez **Exécuter en tant qu'administrateur**.

Pour mettre à jour le centre de documentation DB2 installé sur votre ordinateur ou le serveur intranet, procédez comme suit :

1. Arrêtez le centre de documentation DB2.
 - Sous Windows, cliquez sur **Démarrer** → **Panneau de configuration** → **Outils d'administration** → **Services**. Cliquez ensuite à l'aide du bouton droit de la souris sur le service **Centre documentation DB2** et sélectionnez **Arrêter**.
 - Sous Linux, entrez la commande suivante :
`/etc/init.d/db2icdv97 stop`
2. Démarrez le centre de documentation en mode autonome.
 - Sous Windows :
 - a. Ouvrez une fenêtre de commande.
 - b. Accédez au chemin d'installation du centre de documentation. Par défaut, le centre de documentation DB2 est installé dans le répertoire <Program Files>\IBM\DB2 Information Center\Version 9.7, où <Program Files> représente l'emplacement du répertoire Program Files.
 - c. A partir du répertoire d'installation, accédez au répertoire doc\bin.
 - d. Exécutez le fichier help_start.bat :
`help_start.bat`
 - Sous Linux :
 - a. Accédez au chemin d'installation du centre de documentation. Par défaut, le centre de documentation DB2 se trouve dans le répertoire /opt/ibm/db2ic/version 9.7.
 - b. A partir du répertoire d'installation, accédez au répertoire doc/bin.
 - c. Exécutez le script help_start :
`help_start`

Le navigateur Web par défaut du système ouvre le centre de documentation autonome.

3. Cliquez sur le bouton **Mise à jour** (🔄). (JavaScript™ doit être activé dans votre navigateur.) Sur le panneau droit du centre de documentation, cliquez sur **Rechercher des mises à jour**. Une liste des mises à jour des documentations existantes s'affiche.
4. Pour lancer le processus d'installation, cochez les éléments voulus, puis cliquez sur **Installer les mises à jour**.
5. Une fois le processus d'installation complété, cliquez sur **Terminer**.
6. Arrêtez le centre de documentation autonome :
 - Sous Windows, accédez au répertoire doc\bin du répertoire d'installation et exécutez le fichier help_end.bat :
`help_end.bat`

Remarque : Le fichier help_end contient les commandes requises afin d'interrompre sans risque les processus démarrés par le fichier de commandes help_start. N'utilisez pas Ctrl-C ou toute autre méthode pour interrompre help_start.bat.

- Sous Linux, accédez au répertoire doc/bin du répertoire d'installation et exécutez le script help_end :
`help_end`

Remarque : Le script help_end contient les commandes requises afin d'interrompre sans risque les processus démarrés par le script help_start. N'utilisez pas d'autre méthode pour interrompre le script help_start.

7. Redémarrez le centre de documentation DB2.

- Sous Windows, cliquez sur **Démarrer** → **Panneau de configuration** → **Outils d'administration** → **Services**. Cliquez ensuite à l'aide du bouton droit de la souris sur le **Centre de documentation DB2** et sélectionnez **Démarrer**.
- Sous Linux, entrez la commande suivante :

```
/etc/init.d/db2icdv97 start
```

Le centre de documentation DB2 mis à jour affiche les nouvelles rubriques et les rubriques mises à jour.

Tutoriels DB2

Les tutoriels DB2 présentent différents aspects des produits DB2. Chaque leçon fournit des instructions étape par étape.

Avant de commencer

Vous pouvez consulter la version XHTML du tutoriel à partir du centre de documentation à l'adresse suivante : <http://publib.boulder.ibm.com/infocenter/db2help/>.

Certaines leçons s'appuient sur des exemples de données ou de codes. Reportez-vous au tutoriel pour obtenir une description des conditions préalables aux tâches qu'il présente.

Tutoriels DB2

Pour afficher le tutoriel, cliquez sur le titre.

«pureXML» dans *pureXML Guide*

Configurez une base de données DB2 pour stocker des données XML et effectuer des opérations de base avec le magasin de données XML natif.

«Visual Explain» dans *Tutoriel Visual Explain*

Analyse, optimisation et ajustement des instructions SQL pour l'optimisation des performances à l'aide de Visual Explain.

Informations relatives à la résolution d'incidents sur DB2

Un grand nombre d'informations concernant l'identification et la résolution d'incidents sont à votre disposition lorsque vous utilisez les produits de bases de données DB2.

Documentation DB2

Les informations relatives à l'identification des incidents sont disponibles dans le document *DB2 Troubleshooting Guide* ou dans la section Database fundamentals du *Centre de documentation DB2*. Vous y trouverez des informations utiles pour identifier et isoler les incidents à l'aide d'outils et d'utilitaires de diagnostic DB2, pour résoudre les incidents les plus courants et tout autre incident découlant de l'utilisation de vos produits de base de données DB2.

Site Web de support technique DB2

Reportez-vous au site Web de support technique DB2 si vous rencontrez des incidents et souhaitez être aidé pour en déterminer les causes et pour les résoudre. Le site Web du support technique vous permet d'accéder aux dernières mises à jour des publications DB2, des notes techniques, des enregistrements de correctifs APAR (APAR ou correctifs) et des groupes de correctifs, ainsi qu'à d'autres ressources. Vous pouvez effectuer des recherches dans cette base de connaissances pour trouver d'éventuelles solutions à vos problèmes.

Accédez au site Web de support technique DB2 à l'adresse suivante :
http://www.ibm.com/software/data/db2/support/db2_9/

Dispositions

Les droits d'utilisation relatifs à ces publications sont soumis aux dispositions suivantes.

Usage personnel : Vous pouvez reproduire ces publications pour votre usage personnel, non commercial, sous réserve que toutes les mentions de propriété soient conservées. Vous ne pouvez distribuer ou publier tout ou partie de ces publications ou en faire des oeuvres dérivées sans le consentement exprès d'IBM.

Usage commercial : Vous pouvez reproduire, distribuer et publier ces publications uniquement au sein de votre entreprise, sous réserve que toutes les mentions de propriété soient conservées. Vous ne pouvez reproduire, distribuer, afficher ou publier tout ou partie de ces publications en dehors de votre entreprise, ou en faire des oeuvres dérivées, sans le consentement exprès d'IBM.

Excepté les droits d'utilisation expressément accordés dans ce document, aucun autre droit, licence ou autorisation, implicite ou explicite, n'est accordé pour ces publications ou autres informations, données, logiciels ou droits de propriété intellectuelle contenus dans ces publications.

IBM se réserve le droit de retirer les autorisations accordées ici si, à sa discrétion, l'utilisation des publications s'avère préjudiciable à ses intérêts ou que, selon son appréciation, les instructions n'ont pas été respectées.

Vous ne pouvez télécharger, exporter ou réexporter ces informations qu'en total accord avec toutes les lois et règlements applicables dans votre pays, y compris les lois et règlements américains relatifs à l'exportation.

IBM N'OCTROIE AUCUNE GARANTIE SUR LE CONTENU DE CES PUBLICATIONS. LES PUBLICATIONS SONT LIVREES EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES PUBLICATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Annexe B. Remarques

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations
IBM Canada Ltd.
3600 Steeles Avenue East
Markham, Ontario
L3R 9Z7
Canada

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit à l'adresse suivante :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
3-2-12, Roppongi, Minato-ku,
Tokyo 106-8711 Japan

Le paragraphe suivant ne s'applique ni au Royaume-Uni ni dans aucun autre pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE «EN L'ETAT». IBM DECLINE TOUTE RESPONSABILITE, EXPRESSE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE QUALITE MARCHANDE OU D'ADAPTATION A VOS BESOINS. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'ICA, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

LICENCE DE COPYRIGHT :

Le présent logiciel contient des exemples de programme d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Ces exemples de programmes sont fournis "en l'état", sans garantie d'aucune sorte. IBM ne sera en aucun cas responsable des dommages liés à l'utilisation de ces programmes.

Toute copie totale ou partielle de ces programmes exemples et des oeuvres qui en sont dérivées doit comprendre une notice de copyright, libellée comme suit :

© (*nom de votre société*) (*année*). Des segments de code sont dérivés des Programmes exemples d'IBM Corp. © Copyright IBM Corp. *_indiquez l'année ou les années_*. All rights reserved.

Marques

IBM, le logo IBM et ibm.com sont des marques d'International Business Machines Corp. dans diverses juridictions de par le monde. Les autres noms de produits et de services peuvent appartenir à IBM ou à des tiers. La liste actualisée de toutes les marques IBM est disponible sur la page Web "Copyright and trademark information" à l'adresse www.ibm.com/legal/copytrade.shtml.

Les termes qui suivent sont des marques d'autres sociétés :

- Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.
- Java ainsi que tous les logos et toutes les marques incluant Java sont des marques de Sun Microsystems, Inc. aux Etats-Unis et/ou dans certains autres pays.
- UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.
- Intel, le logo Intel, Intel Inside[®], le logo Intel Inside, Intel[®] Centrino[®], le logo Intel Centrino, Celeron[®], Intel[®] Xeon[®], Intel SpeedStep[®], Itanium[®] et Pentium[®] sont des marques d'Intel Corporation ou de ses filiales aux Etats-Unis et/ou dans certains autres pays.
- Microsoft, Windows, Windows NT[®] et le logo Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

Les autres noms de sociétés, de produits et de services peuvent appartenir à des tiers.

Index

A

- à propos de ce manuel ix
- ACCRDB (commande) 163
- ACCRDBRM (commande) 163
- ACCSEC (commande) 163
- actions recommandées
 - affichage, soumission, sauvegarde 83
- affinités client
 - configuration 133
- aide
 - configuration de la langue 184
 - instructions SQL 183
- alertes d'état 81
- alias de base de données client 76
- améliorations apportées à la version
 - version précédente
 - DB2 Connect version 8 3
- applications
 - conception 105
 - définition des accès 55
 - performances 105
 - procédures mémorisées 105
 - SQL composé 105
 - Web
 - au moyen de DB2 Connect 20
- applications client
 - ID 76
 - rétablissement de la communication 93
- applications Web
 - DB2 Connect 20
 - procédures mémorisées 23
- architecture CDRA (Character Data Representation Architecture) 10
- Architecture de base de données relationnelle répartie (DRDA)
 - accès aux données 9
 - Présentation 9, 10
- arrêt
 - Moniteur de santé DB2 pour z/OS 82
- Assistant de configuration de mise à jour multisite 60
- assistants
 - mise à jour multisite 60
- attributs du serveur d'échanges de données (commande) 163
- authentification 39
 - généralités 50
 - types
 - CLIENT 50, 52
 - DATA_ENCRYPT 50
 - KERBEROS 50
 - SERVER 50
 - SERVER_ENCRYPT 50
 - SERVER_ENCRYPT_AES 50
 - valeur par défaut 50
 - validation 50
- Authentification
 - DB2 Connect
 - support 53
 - REVOKE (instruction) 53

B

- bases de données
 - alias
 - feuille de travail de personnalisation du répertoire 39
 - répertoire système des bases de données 33
 - bases de données hôte 4
 - noms
 - feuille de travail de personnalisation du répertoire 39
 - objet RDBNAM 163
 - répertoire DCS 35
 - répertoire système des bases de données 33
 - optimisation 141
 - outils de mesure des performances 99
 - regroupement de requêtes 105
- bases de données cible
 - nom 35, 39
- bases de données fédérées
 - requête répartie 13
- bases de données hôte
 - accès via DB2 Connect Personal Edition 15
 - connectivité
 - équilibre de la charge 91
 - haut niveau de disponibilité 91
- blocs de requête supplémentaires
 - EXTRA BLOCKS SRV (paramètre) 144
 - présentation 144
- blocs de requêtes
 - augmentation des taux de transfert des données DB2 Connect 144

C

- CCSID (ID de jeu de caractères codés)
 - support des CCSID bidirectionnels
 - description 35
- Centre de contrôle
 - mises à jour multisites 60
- Centre de documentation
 - mise à jour 185
- centre de documentation DB2
 - affichage dans plusieurs langues 184
 - langues 184
 - mise à jour 186
 - versions 184
- chaînes de paramètres
 - double virgules 35
 - virgules 35
- changement d'échelle des fenêtres
 - extensions RFC-1323 146
- CHAR (type de données)
 - description 147
- CLI (call level interface)
 - applications
 - CURRENTPACKAGESET 52
 - connexions sécurisées 45
 - généralités 149
- CLIENT (type d'authentification)
 - remarques relatives à DB2 Connect 50
- clients
 - activation du support XA 132

- clients (*suite*)
 - équilibrage des charges de travail 124
 - redirection automatique du client 127
 - clients de serveur de données IBM
 - équilibrage de charge de travail 124
 - redirection automatique du client, support 127
 - commande db2trc
 - formatage d'une sortie de trace 159
 - Présentation 157
 - vidage d'une sortie de trace 158
 - commande de manuels DB2 182
 - commande de validation
 - mémoires tampon de la sortie de la trace 163
 - Commande GET SNAPSHOT 74
 - commandes
 - ACCRDB 163
 - ACCRDBRM 163
 - ACCSEC 163
 - commit 163
 - db2drdat 161
 - db2trc 157, 159
 - EXCSAT 163
 - EXCSATRD 163
 - GET SNAPSHOT 74
 - SECCHK 163
 - COMMIT (instruction)
 - liaison statique 105
 - concentrateur de connexion 109
 - concentrateurs de connexion
 - agents 111
 - agents logiques 111
 - comparaison avec le regroupement de connexions 116
 - DB2 Connect 117
 - exemples 111
 - généralités 109
 - MAX_COORDAGENTS (paramètre de configuration) 111
 - mise en oeuvre 111
 - NUM_INITAGENTS (paramètre de configuration) 111
 - NUM_POOLAGENTS (paramètre de configuration) 111
 - paramètres de configuration 111
 - regroupement 111
 - répartiteur 111
 - restrictions 111
 - support des transactions XA 111
 - temps système 111
 - configuration
 - connexions hôte 15
 - remarques, changement du mot de passe 52
 - conflit
 - ressources système 143
 - connectivité
 - serveurs, DB2 Connect Enterprise Edition 19
 - connexions
 - concentrateurs, voir concentrateurs de connexions 111
 - connexion directe à un hôte 15
 - connexion directe avec grand système IBM 15
 - DB2 Connect Enterprise Edition 19
 - directes à l'hôte System z ou à la base de données IBM i 17
 - regroupement
 - avantages 111
 - concentrateurs de connexion 111
 - généralités 109
 - rétablissement
 - connexion directe à un hôte 15
 - DB2 Connect Enterprise Edition 19
 - connexions sécurisées 45
 - connexions sécurisées (*suite*)
 - changement d'utilisateurs via CLI/ODBC 48
 - via CLI/ODBC 46
 - contexte sécurisé
 - support DB2 Connect 45
 - via CLI/ODBC 46
 - contrôle
 - connexions
 - Serveur DB2 Connect 73
 - moniteur de performances Windows 73
 - conversion de données
 - hôte 146
 - conversions
 - données hôte 146
 - CURRENTPACKAGESET (mot-clé CLI/ODBC) 52
- ## D
- DATA_ENCRYPT (type d'authentification) 50
 - dates
 - support des décalages horaires 35
 - DB2 Connect
 - améliorations
 - fonctions, 3
 - concentrateurs de connexion 117
 - Configuration
 - grand système IBM 62
 - IBM Power Systems 62
 - System z 62
 - Enterprise Edition
 - API 22
 - applications Web 20
 - gestionnaires de transactions compatibles XA 62
 - JDBC 22
 - moniteurs de traitement des transactions 27
 - serveurs de connectivité 19
 - serveurs Web 23
 - SQLJ 22
 - généralités 3
 - prise en charge grand système IBM 15
 - prise en charge System i 15
 - produits 3
 - sécurité 45
 - serveur de connectivité, scénarios 15
 - support de l'environnement Sysplex 119
 - support hôte 15
 - support Sysplex pour les clients 123
 - transfert de données 63
 - DB2 Universal Database pour OS/390 et z/OS
 - DOMAIN 34
 - DYNAMICRULES (BIND) (option) 52
 - ensemble de données d'amorçage 34
 - paramètres BSDS 34
 - RESPORT 34
 - sécurité 52
 - TCPPORT 34
 - valeurs du répertoire des noeuds 34
 - db2drdat (commande)
 - fichier de sortie 161
 - dcs1ari.map (fichier) 67
 - dcs1dsn.map (fichier) 67
 - dcs1qsq.map (fichier) 67
 - ddcstrc (utilitaire) 162
 - DDM (Distributed Data Management)
 - architecture DRDA (Distributed Relational Database Architecture) 10

- DDM (Distributed Data Management)
 - sortie db2drdat 161
- débit
 - transactions 99
- décimaux non condensés
 - types de données 146
- définition des accès
 - droits
 - marqueurs de paramètre avec décalage 55
 - modules 55
 - noms de module 55
 - utilitaires et applications 55
- demandeurs d'application (AR)
 - définition de l'architecture DRDA 10
 - paramètres 39
- démarrage
 - Moniteur de santé DB2 pour z/OS 82
- DESCRIBE (instruction)
 - instructions SQL composées 105
 - performances avec l'instruction PREPARE 105
- développement d'applications 105
 - client DB2 AD 15
 - ODBC 15
- DIRCACHE (paramètre) 139
- dispositions
 - utilisation des publications 189
- Distributed Data Management (DDM)
 - Architecture de base de données relationnelle répartie (DRDA) 10
 - sortie db2drdat 161
- documentation
 - conditions d'utilisation 189
 - imprimés 180
 - PDF 180
 - présentation 179
- données
 - flots
 - DB2 Connect 10, 99
 - groupage 105
 - sources
 - requête répartie 13
 - transfert
 - débits 99, 147
 - entre hôtes et postes de travail 63
 - performances 147
- données de diagnostic
 - présentation 156
- données de type caractère 147
- droit CREATE IN COLLECTION NULLID 55
- droits d'accès
 - définition des accès 55

E

- échecs de connexion
 - redirection automatique de client 95
- enableDirectXA 132
- équilibre de charge de travail
 - configuration pour les clients 125
 - côté client 124
- erreurs
 - résolution des incidents 153
- état du système
 - Commande GET SNAPSHOT 74
- EXCSAT (commande) 163
- EXCSATRD (commande) 163

- EXECUTE IMMEDIATE (instruction)
 - conception d'applications 105
- exemples
 - concentrateurs de connexion 111
 - concentrateurs XA 111
- EXTNAM (objet) 163

F

- feuilles de travail
 - personnalisation des répertoires 39
- fichier ddcs400.lst 55
- fichier ddcsmv.lst 55
- fichier ddcsvm.lst 55
- fichier ddcsvse.lst 55
- fichier de configuration db2dsdriver
 - paramètres de l'équilibrage de charge de travail 125
- fichier de mappage SQLCODE 67
- fichiers core
 - identification des incidents 156
- fonction de contrôle d'accès aux données (RACF)
 - Authentification 53
- fonction de trace
 - DB2 traces 157, 158, 159
 - Présentation 157
 - traces DRDA 164, 169
- FOR FETCH ONLY (clause)
 - SELECT (instruction) 105
- FORCE (commande)
 - ID agent 76
- format décimal condensé 146
- Formatted Data Object Content Architecture (FDOCA) 10
- fuseaux horaires
 - présentation 35

G

- gestion des connexions 109
- gestionnaire de points de synchronisation (SPM)
 - paramètres par défaut 62
 - scénarios 61
- gestionnaires de transactions XA
 - concentrateurs de connexion 111
 - description 27
- goulots d'étranglement
 - performances 99
 - transactions 99
- GRANT (instruction)
 - sécurité 53
- groupage de données 105

H

- haut niveau de disponibilité
 - DB2 Connect 89

I

- IBM i
 - DRDA 10
- IBM WebSphere 21
- ID application hôte 76
- ID autorisation 76
- ID de page de codes 76
- ID produit client 76

- identification des incidents
 - après connexion 154
 - collecte des informations 153
 - connexion 153, 154
 - DB2 Connect 173
 - fonctions de trace 157
 - DRDA 164, 169
 - informations disponibles 188
 - outils de diagnostic
 - présentation 156
 - performances 143
 - tutoriels 188
- InfoSphere Federation Server
 - présentation 7
- instructions
 - COMMIT 105
 - DESCRIBE 105
 - EXECUTE IMMEDIATE 105
 - FOR FETCH ONLY 105
 - GRANT 53
 - PREPARE 105
 - REVOKE 53
 - ROLLBACK
 - conception d'applications 105
 - SELECT 105
- instructions SQL
 - affichage de l'aide 183
 - DB2 Connect 5
- interpréteur de commandes
 - instructions SQL 6
 - performances 105
- INTERRUPT_ENABLED (déconnexion), paramètre 35

J

- Java
 - serveurs d'applications
 - API 22
 - DB2 Connect 22
 - JDBC 22
 - SQLJ 22
- jetons
 - SQLCODE 67
- journal d'évaluation des règles
 - Moniteur de santé DB2 pour z/OS 81
- journaux
 - évaluation de règle 81

L

- LIST DCS APPLICATIONS (commande) 76
- liste d'adresses placées dans la mémoire cache 120
- liste de liens 55
- LOCALDATE (paramètre) 35

M

- manuels
 - imprimés
 - commande 182
- mappage
 - SQLCODE 67
 - NOMAP (paramètre) 67
- mappage des SQLCODE
 - adaptation 67
 - NOMAP (paramètre) 67

- matériel
 - performances du réseau 147
- mémoire tampon d'envoi
 - données de trace 161
- mémoire tampon de réception 161
- message de réponse indiquant la fin d'une unité d'oeuvre (ENDUOWRM) 163
- messages d'erreurs
 - DB2 Connect 173
- Microsoft Windows
 - applications 15
- mise en mémoire cache des répertoires (paramètre de configuration)
 - Optimisation de DB2 Connect 139
- mises à jour
 - Centre de documentation 185
 - centre de documentation DB2 186
 - répertoires de base de données 33
- mises à jour multisites
 - activation 59
 - Centre de contrôle 60
 - gestionnaire de points de synchronisation 61
 - test 60
 - unité d'oeuvre répartie (DUOW) 59
- modèle DTP (Distributed Transaction Processing) X/Open 27
- modules
 - serveurs de base de données hôte 55
 - serveurs de base de données System i 55
- moniteur de santé
 - DB2 pour z/OS 81
- Moniteur de santé DB2 for z/OS
 - Présentation 81
- Moniteur de santé DB2 pour z/OS
 - affichage, soumission, sauvegarde des actions recommandées 83
 - affichage des objets d'alerte 87
 - affichage des récapitulatifs d'alerte 86
 - démarrage, arrêt, régénération 82
- moniteur du gestionnaire de bases de données
 - clients éloignés 73
 - Présentation 6
- moniteurs de traitement des transactions
 - caractéristiques 27
 - exemples 27
 - mises à jour multisites 59
 - OLTP 27
 - transactions 27
 - Tuxedo 27
- mots de passe
 - modification
 - OS/390 et z/OS 52

N

- NNAME client 76
- noeuds
 - nom
 - feuille de travail de personnalisation du répertoire 39
 - valeurs du répertoire des noeuds 34
 - valeurs du système de base de données 33
 - répertoires
 - mise à jour 33
 - valeurs 34
- nom de base de données cible AS 35
- nom de destination symbolique 39
 - distinction majuscules/minuscules 34
- nom de l'application (élément de contrôle) 76

- NOMAP (paramètre)
 - désactivation du mappage SQL 67
 - mappage SQLCODE 35
 - paramètres du répertoire DCS 67
- NULLID
 - OS/400 55
- NUMDB
 - paramètre 139
- numéro de séquence client 76
- numéro de séquence sortante
 - DB2 LIST DCS APPLICATIONS (commande) 76

O

- objet SRVNAM 163
- objets d'alerte
 - affichage 87
- ODBC (Open Database Connectivity)
 - applications
 - CURRENTPACKAGESET 52
 - généralités 149
 - interface 15
 - optimisation de l'accès 103
- optimisation
 - DB2 Connect 97
 - DB2 for OS/390 and z/OS 144
 - paramètres
 - AGENTPRI 139
 - DIRCACHE 139
 - MAXAGENTS 139
 - MAXDARI 139
 - NUMDB 139
 - RQRIOBLK 139
 - performances
 - base de données 141
 - réseau 141
- OS/390
 - DRDA 10
- outils
 - performances 99
 - utilisation de l'UC 99
 - utilisation de la mémoire 99
- outils d'exploitation de la mémoire 99

P

- paramètre AGENTPRI, configuration du gestionnaire de base de données 139
- paramètre D (déconnexion) 35
- paramètre db2dsdriver AFFINITY_LIST 133
- paramètre db2dsdriver ALTERNATE_SERVER_LIST 133
- paramètre db2dsdriver CLIENT_AFFINITY 133
- paramètre de configuration MAXDARI 139
- paramètre enableAcr 129
- paramètre enableSeamlessAcr 129
- paramètre enableWLB 125
- paramètre MAX_COORDAGENTS, configuration du gestionnaire de base de données 109, 111
- paramètre MAXAGENTS, configuration du gestionnaire de base de données 139
- paramètre maxRefreshInterval 125
- paramètre maxTransportIdleTime 125
- paramètre maxTransports 125
- paramètre maxTransportWaitTime 125
- paramètre NUM_INITAGENTS, configuration du gestionnaire de base de données 109, 111

- paramètre NUM_POOLAGENTS, configuration du gestionnaire de base de données 109, 111
- paramètres
 - chaînes 41
 - MAXAGENTS 139
 - PRDID 163
 - répertoires 39
 - SYSPLEX 35
- paramètres d'ensemble de données d'amorçage (BSDS)
 - Z/OS et OS/390 34
- paramètres de configuration
 - AGENTPRI 139
 - DIRCACHE 139
 - MAX_COORDAGENTS 109, 111
 - MAXDARI 139
 - NUM_INITAGENTS 109, 111
 - NUM_POOLAGENTS 109, 111
 - NUMDB 139
 - RQRIOBLK 139
 - TCP_KEEPALIVE 95
- performances
 - accès ODBC 103
 - applications 105
 - concentrateur de connexion 116
 - DB2 Connect
 - augmentation des taux de transfert 144
 - identification des incidents 143
 - optimisation 97
 - présentation 99
 - impact de l'interpréteur de commandes 105
 - OS/390 144
 - regroupement de connexions 116
 - réseau
 - matériel 147
 - ressources système 143
 - z/OS 144
- perluète (double &)
 - fichier de mappage SQLCODE 67
- personnalisation
 - répertoires, feuilles de travail associées 39
- pilotes de serveur de données IBM
 - équilibre de charge de travail 124
 - redirection automatique du client, support 127
- pool de transport 124
- PRDID (paramètre) 163
- prédicats
 - performances de la logique 105
- PREPARE (instruction)
 - conception d'applications 105
 - effet sur les performances 105
- privilège BINDADD
 - droits requis pour la définition des accès 55
- procédures mémorisées
 - généralités 23
- programmation CGI (Common Gateway Interface)
 - avantages 20
 - limitations 20
- programmation CGI (Common Gateway Interface, Interface de passerelle commune)
 - avantages 20
 - limitations 20
- protocole d'authentification Kerberos
 - DB2 Connect 50
 - OS/390 et z/OS 51
- ps (commande)
 - EXTNAM (objet) 163
 - présentation 156

R

- récapitulatifs des alertes
 - affichage 86
- recommandations 191
- redirection automatique de client
 - échecs de connexion 95
- redirection automatique du client
 - configuration pour les clients 129
 - côté client 127
- redirection client automatique
 - configuration pour DB2 Connect 93
 - description 93
- références
 - définition de plusieurs entrées de base de données 40
- régénération
 - Moniteur de santé DB2 pour z/OS 82
- regroupement de connexions 109
 - comparaison avec le concentrateur de connexion 116
 - généralités 109
- Relational Connect
 - description du produit 7
- relations de confiance
 - contextes sécurisés et connexions sécurisées 45
- répertoire DCS (Database Connection Services)
 - mise à jour d'entrées 33
 - valeurs 35
 - voir répertoire DCS (Database Connection Services) 35
- répertoire système des bases de données
 - mise à jour 33
 - valeurs 33
- répertoires
 - base de données système
 - mise à jour 33
 - valeurs 33
 - feuilles de travail de personnalisation 39
- répertoires de base de données
 - Database Connection Services (DCS) 33
 - entrées multiples 40
 - mise à jour 33
 - noeud 33
- requêtes de la base de données
 - groupement pour l'amélioration des performances 105
- requêtes réparties
 - bases de données fédérées 13
 - compensation 13
 - présentation 13
 - support DB2 Connect 13
 - transparence des emplacements 13
- réseau
 - optimisation 141
 - outils de mesure des performances 99
 - vitesse de transfert des données 147
- résolution des incidents
 - généralités
 - DB2 Connect 153
 - informations en ligne 188
 - tutoriels 188
- ressources système
 - conflit 143
- restrictions
 - concentrateur de connexion 111
- rétablissement de la communication
 - applications client 93
- REVOKE (instruction)
 - Authentification 53
- ROLLBACK (instruction)
 - liaison statique 105

- RQRIOLBK (paramètre)
 - optimisation 139

S

- scénarios
 - sécurité TCP/IP 53
- SECCHK (commande) 163
- sécurité
 - codes étendus
 - OS/390 et z/OS 52
 - conseils 52
 - GRANT (instruction) 53
 - INTRODUCTION 45
 - Kerberos 51
 - TCP/IP 53
 - types 39
 - valeurs du répertoire des noeuds 34
- SELECT (instruction)
 - dans la conception d'une application 105
 - FOR FETCH ONLY 105
 - modifiables 105
- SERVER (type d'authentification) 50
- SERVER_ENCRYPT (type d'authentification) 50
- serveurs
 - application
 - DB2 Connect EE 23
- serveurs d'applications (AS)
 - clients lourds 23
 - configuration 23
 - définition de l'architecture DRDA 10
 - déploiement 23
 - généralités 23
 - modèles à 2 niveaux et 3 niveaux 23
 - support DB2 Connect 23
- serveurs Web
 - DB2 Connect 23
- SET CURRENT PACKAGESET (instruction) 52
- SHOW DETAIL (option du moniteur) 76
- SOCKS
 - noeud
 - variables d'environnement obligatoires 34
- SQL (Structured Query Language)
 - dynamique 105
 - statique 105
- SQL_ATTR_
 - TRUSTED_CONTEXT_PASSWORD
 - utilisation 48
 - TRUSTED_CONTEXT_USERID
 - utilisation 48
 - USE_TRUSTED_CONTEXT
 - utilisation 46
- SQL composé
 - NON ATOMIQUE 105
- SQL composé ATOMIQUE
 - non pris en charge dans DB2 Connect 105
- SQL composé NON ATOMIQUE
 - conception d'applications 105
- SQL/DS
 - DRDA 10
- SQL dynamique
 - CURRENTPACKAGESET CLI/ODBC (paramètre de configuration) 52
 - effets du traitement 5
 - performances
 - techniques 105

- SQL statique
 - effets du traitement 5
 - performances 105
- SQL0965 (code d'erreur) 173
- SQL0969 (code d'erreur) 173
- SQL1338 (code d'erreur) 34, 173
- SQL30020 (code d'erreur) 173
- SQL30060 (code d'erreur) 173
- SQL30061 (code d'erreur) 173
- SQL30073 (code d'erreur) 173
- SQL30081N (code d'erreur) 173
- SQL30082 (code d'erreur) 173
- SQL5043N (code d'erreur) 173
- SQLCODE
 - mappage 67
 - zone dans SQLCA 161
- SQLDA (SQL Descriptor Area)
 - taille allouée 105
- SQLSTATE
 - codes de classe 67
- START MVS (commande système) 81
- STOP MVS (commande système) 81
- structure SQLCA (SQL Communication Area)
 - mémoires tampon de données 161
 - SQLCODE (zone) 161
- support CCSID bidirectionnel
 - BIDI (paramètre) 35
- support du Sysplex client
 - limitations 138
- support XA
 - activation côté clients 132
- surveillance
 - connexions 73
 - performances 73
- Sysplex
 - configuration requise 121
 - considérations concernant System z 119
 - équilibre de la charge 120
 - informations sur les niveaux de priorité 120
 - limitations pour le support côté client 138
 - paramètre 35
 - support client 123
 - support DB2 Connect 119
 - tolérance aux pannes 120
 - utilisation 120
- System i
 - DRDA 10
- système d'aide à la décision 161
- systèmes d'exploitation Windows
 - Moniteur de performances
 - contrôle des applications DB2 73
- systèmes Power
 - DRDA 10

T

- taille de bloc 139
- taille du bloc de pagination 139
- TCP/IP
 - ACCSEC (commande) 163
 - Authentification
 - scénarios 53
 - configuration
 - connexions hôte 17
 - DOMAIN 34
 - extensions RFC-1323
 - changement d'échelle des fenêtres 146
 - TCP/IP (suite)
 - noms d'hôte 39
 - noms d'hôtes éloignés 34, 39
 - noms de service 34
 - numéros de ports 39
 - port utilisé pour la resynchronisation 34
 - RESPORT 34
 - SECCHK (commande) 163
 - sécurité
 - vérification 52
 - TCPPTORT 34
 - TCP_KEEPALIVE
 - paramètre de configuration du système d'exploitation 95
 - temps de réponse 99
 - test
 - mises à jour multisites 60
 - tests de performances
 - performances 99
 - traces
 - données entre DB2 Connect et le serveur 161
 - DRDA
 - interprétation 161
 - exemples de fichier de sortie 164
 - fichier de sortie 161, 162
 - informations de mémoire tampon pour les traces
 - DRDA 169
 - transactions
 - applications réparties XA 62
 - DB2 Connect Enterprise Edition 27
 - débit 99
 - mises à jour multisites 9, 59
 - moniteurs de traitement des transactions 27
 - réparties
 - serveurs pris en charge 59
 - support 62
 - unité d'oeuvre (UOW) 9
 - validation en deux phases 9
 - transactions distribuées
 - client support for XA 131
 - transfert de données
 - au moyen de DB2 Connect 63
 - transports 124
 - tutoriels
 - identification des incidents 188
 - résolution des incidents 188
 - Visual Explain 188
 - Tuxedo
 - DB2 Connect Enterprise Edition 27
 - type d'authentification PROGRAM 53
 - type d'authentification SAME 53
 - type d'authentification SERVER_ENCRYPT_AES 50
 - type de données INTEGER 146
 - types d'authentification NONE 53
 - types de données
 - caractères 147
 - CHAR 147
 - conversion
 - effet sur les performances 146
 - décimaux non condensés 146
 - format décimal condensé 146
 - INTEGER
 - conversion de données sur l'hôte 146
 - VARCHAR
 - Présentation 147
 - virgule flottante
 - conversion de données sur l'hôte 146

U

- unité centrale
 - outils d'exploitation 99
- unité d'oeuvre éloignée
 - caractéristiques 11
 - exemple 11
 - généralités 11
- unité d'oeuvre répartie
 - mises à jour multisites 59
 - Présentation 9
 - serveurs pris en charge 59
 - validation en deux phases 59
- unités d'oeuvre (UOW)
 - à distance 11
 - définition 9
 - réparties 59
- utilitaire d'administration
 - DB2 Connect 6
- utilitaire d'exportation
 - transfert de données entre hôtes et postes de travail 63
- utilitaire d'importation
 - transfert de données entre un hôte et un poste de travail 63
- utilitaire de contrôle de l'état des processus 156, 163
- utilitaire de trace (db2drdat) 161
- utilitaires
 - administration, DB2 Connect 6
 - db2drdat 161
 - ddcspkgn 55
 - définition des accès 55
 - état des processus 163
 - moniteur du gestionnaire de bases de données 6
 - ps (process status), état des processus 156, 163
 - trace 161

V

- valeur d'authentification 33
- Valeurs du répertoire DCS 35
- validation en deux phases
 - activation 59
 - port de resynchronisation utilisé par les connexions TCP/IP 34
- VARCHAR (type de données)
 - description 147
- vidage d'une trace vers un fichier
 - Présentation 158
- virgule flottante
 - type de données 146
- Virtual Telecommunications Access Method (VTAM) 53
- Visual Explain
 - tutoriel 188
- VM
 - DRDA
 - avec DB2 Connect 10
- VSE
 - DRDA 10

W

- WebSphere
 - généralités 21
- WebSphere MQ
 - DB2 Connect 117

X

- XA
 - connexions sécurisées 45
 - exemples de concentrateur 111
 - gestionnaires de ressources 27
 - support côté client 131

Z

- z/OS
 - DRDA 10



SC11-6572-00



Spine information:

IBM DB2 Connect 9.7

DB2 Connect - Guide d'utilisation

