



Guide de l'utilisateur



Guide de l'utilisateur

Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section *Remarques*.

Remarque

Les captures d'écran de ce manuel ne sont pas disponibles en français à la date d'impression.

Première édition - juillet 2006

Réf. US : GC10-4229-00

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 CONTREFACON 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 2006. Tous droits réservés.

© **Copyright International Business Machines Corporation 1993, 2006. All rights reserved.**

Table des matières

A propos de ce manuel	v
Public concerné	v

Partie 1. Concept DB2 Connect . . . 1

Chapitre 1. Concepts DB2 Connect . . . 3

DB2 Connect	3
Offres de produits DB2 Connect	3
Fonctions fournies avec la version 9 et les versions précédentes.	4
Bases de données hôte	6
DB2 Connect et instructions SQL	7
Utilitaires d'administration DB2 Connect	8
WebSphere Federation Server et DB2 Connect	9

Chapitre 2. Architecture de base de données relationnelle répartie (DRDA) . 11

Architecture de base de données relationnelle répartie (DRDA).	11
DRDA et accès aux données	11
DB2 Connect et DRDA	12
Unité d'oeuvre éloignée	13
Requêtes réparties	15

Chapitre 3. Scénarios DB2 Connect . . 17

Scénarios DB2 Connect	17
Scénarios	17
Accès direct aux bases de données hôte	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	24
DB2 Connect et moniteurs de traitement de transactions	27

Partie 2. Référence 31

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

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	40
Définition d'entrées multiples pour la même base de données	41
Traitement des données bidirectionnelles (BiDi)	42

Chapitre 5. Sécurité 45

Considérations sur l'authentification DB2	45
Support Kerberos	46
Connexions sécurisées	47
Connexions sécurisées via DB2 Connect	47
Création et arrêt d'une connexion sécurisée via CLI	49
Changement d'utilisateur sur une connexion sécurisée via CLI	50
Considérations de sécurité DB2 Connect applicables à DB2 pour OS/390 et z/OS.	53
Astuces et conseils supplémentaires pour la sécurité OS/390 et z/OS	53
Types de sécurité pris en charge avec DB2 Connect	54

Chapitre 6. Utilitaires et applications de liaison 57

Applications et utilitaires de liaison (DB2 Connect)	57
--	----

Chapitre 7. Mises à jour multisites . . . 61

Mises à jour multisites.	61
Activation des mises à jour multisites à l'aide du Centre de contrôle	62
Test d'une mise à jour multisite à l'aide du Centre de contrôle	63
Mise à jour multisite et gestionnaire de points de synchronisation (SPM).	63
Configuration de DB2 Connect avec un gestionnaire de transactions compatible avec XA	64
Support DB2 Connect pour les transactions à couplage lâche	65

Chapitre 8. Mappage SQLCODE 67

Mappage SQLCODE	67
Désactivation du mappage de codes SQLCODE	67
Personnalisation du mappage SQLCODE	67

Chapitre 9. Moniteur du gestionnaire de base de données 73

Contrôle des connexions des clients éloignés	73
Contrôle des performances à l'aide du moniteur de performances de Windows	74
Utilisation des commandes GET SNAPSHOT	75
Etat de l'application DCS.	77

Chapitre 10. Haute disponibilité 83

Haute disponibilité et équilibrage de la charge de travail pour la connectivité de la base de données hôte	83
Redirection automatique du client - Description et configuration	84
Remarques du distributeur	86

Chapitre 11. Performances 89

Considérations sur les performances de DB2 Connect	89
Conception d'application	92
Gestion des connexions	95
Regroupement de connexions	95
Concentrateur de connexion	98
Regroupements de connexions et concentrateur de connexion	103
Support Sysplex DB2 Connect	103
Support Sysplex DB2 Connect	103
Considérations sur l'exploitation de OS/390 et zSeries SYSPLEX	104
Configuration requise pour Sysplex	105
Exploitation de Sysplex avec DB2	105
Optimisation de DB2 Connect	107
Optimisation de DB2 Connect	107
Optimisation de la base de données hôte	109
Considérations relatives à l'optimisation du réseau	109
Conflit des ressources système	111
Résolution des incidents de performances de DB2 Connect	112
Optimisation de DB2 pour OS/390 et z/OS	112
Optimisation de l'accès ODBC	113
Optimisation des performances d'applications CLI/ODBC	114
Augmentation des taux de transfert de données DB2 Connect	115
Bloc de requête supplémentaire	116
Mise à l'échelle des fenêtres RFC-1323	117
Conversion de données sur l'hôte	118
Types de données pour les données de type caractères	118
Matériel réseaux	119

Chapitre 12. Identification des incidents	121
Identification d'incident	121
Concepts de détermination d'incident	121
Regroupement d'informations adéquates	121
Outils de diagnostic	122
Connexion initiale non aboutie	122
Incidents rencontrés après une connexion initiale	123

Utilitaire de trace	124
Caractéristiques de l'utilitaire de trace	125
Sortie de trace	125
Analyse du fichier de sortie de trace	126
Modèles de fichiers de sortie de trace	128
Informations de mémoire tampon postérieures pour les traces DRDA	133
Incidents DB2 Connect communs	134

Partie 3. Annexes 139

Annexe A. Transfert de données à l'aide de DB2 Connect 141

Annexe B. DB2 Database - Informations techniques. 145

Présentation des informations techniques relatives à DB2	145
Commentaires sur la documentation	145
Bibliothèque technique DB2 au format PDF	146
Commande de manuels imprimés DB2	149
Affichage de l'aide sur les codes d'état SQL à partir de l'interpréteur de commandes	150
Accès aux différentes versions du Centre de documentation DB2	150
Affichage des rubriques du Centre de documentation DB2 dans votre langue préférée	150
Mise à jour du Centre de documentation installé sur votre ordinateur ou sur un serveur intranet	151
Tutoriel DB2 Visual Explain	153
Informations concernant la résolution d'incidents liés à DB2	153
Dispositions	154

Annexe C. Remarques 157

Remarques	159
---------------------	-----

Index 161

Comment prendre contact avec IBM 169

A propos de ce manuel

Ce manuel contient des informations d'utilisation générale des produits IBM DB2 Connect suivants :

- DB2 Connect Enterprise Edition
- DB2 Connect Application Server Edition
- DB2 Connect Unlimited Edition pour zSeries
- DB2 Connect Unlimited Edition pour iSeries
- DB2 Connect Personal Edition

Public concerné

Ce manuel s'adresse aux programmeurs et aux administrateurs qui sont responsables de la configuration et de l'entretien des connexions DB2 Connect. Ces connexions sont possibles entre les clients DB2 et l'un des systèmes de gestion de base de données de serveur d'applications suivants :

- DB2 Universal Database (UDB) pour OS/390 et z/OS version 7, et DB2 UDB pour z/OS version 8 ou version ultérieure
- DB2 Server pour VSE & VM Version 7
- DB2 UDB pour iSeries version 5.1 ou version ultérieure
- Autre système de gestion de base de données relationnelle qui implémente une fonction de serveur d'application DRDA.

Remarque : Les applications qui s'exécutent sous z/OS, iSeries ou VM/VSE peuvent accéder aux bases de données DB2 des serveurs Linux, UNIX ou Windows sans DB2 Connect.

Vous trouverez les dernières informations sur DB2 Connect en ligne au Centre de documentation DB2.

Pour accéder au Centre de documentation iSeries, consultez le site Web suivant, <http://www.ibm.com/eserver/iseries/infocenter>.

Partie 1. Concept DB2 Connect

Chapitre 1. Concepts DB2 Connect

DB2 Connect

DB2 Connect offre une connectivité rapide et robuste aux serveurs de base de données hôte et iSeries pour les applications e-business et d'autres applications fonctionnant sur les systèmes d'exploitation Linux, UNIX et Windows.

DB2 Connect Personal Edition fournit une connectivité directe aux serveurs hôte et iSeries DB2 alors que le serveur DB2 Connect offre une connectivité indirecte qui permet aux clients d'accéder aux serveurs hôte ou iSeries DB2 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.

Concepts connexes :

- «DB2 Connect et DRDA», à la page 12
- «Scénarios DB2 Connect», à la page 17

Référence connexe :

- «Offres de produits DB2 Connect», à la page 3

Offres de produits DB2 Connect

DB2 Connect dispose de plusieurs solutions de connexion et notamment de DB2 Connect Personal Edition, ainsi que d'un certain nombre de serveurs DB2 Connect :

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

Pour obtenir des informations détaillées sur les offres de produits DB2 Connect, consultez le site Web suivant <http://www.ibm.com/support/docview.wss?rs=73&uid=swg21219983>

Tâches connexes :

- «Installation d'un serveur DB2 Connect (AIX)» dans *Quick Beginnings for DB2 Connect Servers*
- «Installation d'un serveur DB2 Connect (HP-UX)» dans *Quick Beginnings for DB2 Connect Servers*
- «Installation d'un serveur DB2 Connect (Linux)» dans *Quick Beginnings for DB2 Connect Servers*
- «Installation d'un serveur DB2 Connect (Solaris)» dans *Quick Beginnings for DB2 Connect Servers*
- «Installation d'un serveur DB2 Connect (Windows)» dans *Quick Beginnings for DB2 Connect Servers*
- «Installation de DB2 Connect Personal Edition (sous Linux)» dans *Quick Beginnings for DB2 Connect Personal Edition*

- «Installation de DB2 Connect Personal Edition (sous Windows)» dans *Quick Beginnings for DB2 Connect Personal Edition*

Fonctions fournies avec la version 9 et les versions précédentes

Cette section est un récapitulatif des améliorations introduites dans chaque version et édition.

Fonctions fournies dans DB2 Connect version 9

DB2 Connect version 9 inclut les améliorations suivantes :

- Support client pour les connexions sécurisées
Un client peut créer des connexions sécurisées à l'aide d'ODBC, de XA ou de nouvelles méthodes Java dans des serveurs de base de données (actuellement DB2 pour z/OS) prenant en charge les contextes sécurisés. Le nom d'utilisateur du client peut ensuite être modifié sans que le serveur de base de données n'ait à authentifier complètement le nouveau nom.
- Support de type de données BINARY, VARBINARY et DECFLOAT
DB2 pour z/OS prend désormais en charge les types de données BINARY, VARBINARY et DECFLOAT. La prise en charge de ces types de données a été ajoutée dans DB2 CLI et DB2 .NET Data Provider. Vos applications qui utilisent DB2 Connect pour évaluer DB2 pour z/OS peuvent utiliser DB2 CLI et DB2 .NET Data Provider pour tirer partie des nouveaux types de données. Un nouveau paramètre de connexion SQL_ATTR_DECFLOAT_ROUNDING_MODE permet au client de spécifier le type d'arrondi qui doit être utilisé lorsqu'une opération effectuée côté serveur requiert l'arrondissement d'une valeur décimale.
- Les protocoles de communication NetBIOS et SNA ne sont plus pris en charge
Les clients qui utilisent ces protocoles doivent cataloguer à nouveau leurs noeuds et leurs bases de données à l'aide d'un protocole pris en charge, tel que le protocole TCP/IP.
- Le support du protocole de communications IPv6 a été ajouté
Le support a été ajouté pour le protocole IP version 6 (IPv6). Aussi, vous pouvez désormais vous connecter aux serveurs en utilisant leurs adresses IPv4 ou IPv6.
- La limitation de l'interpréteur de commandes (CLP) à 64 K pour les instructions SQL a été supprimée
Une nouvelle limitation de l'interpréteur de commandes à environ 2 Mo pour les instructions SQL et les commandes CLP contenant des composants d'état SQL est comparable aux limitations définies dans d'autres outils DB2. Vos applications qui utilisent DB2 Connect peuvent désormais bénéficier de cette nouvelle limitation.
- Améliorations apportées à DB2 .NET Data Provider notamment au support .NET Framework 2.0
Ce support et ces améliorations vous aideront à développer des applications .NET plus puissantes en vue d'une utilisation avec DB2 Connect. Parmi les nouvelles fonctions se trouvent :
 - Des applications qui extraient un ensemble spécifique de lignes au lieu de parcourir la totalité d'un ensemble de résultats.
 - Des applications qui effectuent une opération de copie de données non formatées.

- Des applications qui déterminent le nombre d'instructions SQL à rassembler avant d'être utilisées en tant que lot dans le serveur de base de données DB2. Ces fonctions engendrent moins de transmissions de données individuelles entre l'application client et le serveur de base de données.
- La validation en deux phases des sources de données multifournisseur lorsque WebSphere Federation Server est utilisé
Les applications DB2 Connect peuvent utiliser WebSphere Federation Server pour atteindre des sources de données offertes par les fournisseurs IBM et non IBM.
- Support du délai de connexion pour les applications de base de données
Vous pouvez limiter le délai d'attente d'obtention d'une application de vos applications de base de données DB2 Connect. Ce point est particulièrement important lorsque le serveur de base de données est inaccessible.
- Mise à niveau de DB2 Connect Personal Edition plus facile
Vous pouvez mettre à jour DB2 Connect Personal Edition sur les systèmes d'exploitation Windows et Linux en fournissant le fichier de certification électronique adéquat. Il n'est plus nécessaire d'effectuer une installation complète lorsque vous procédez à la mise à niveau.
- Modifications apportées au support de licences DB2
Les modifications du conditionnement DB2 Connect font partie des améliorations apportées à la commande du Centre de gestion des licences et de l'outil de gestion sous licence (**db2licm**).

Fonctions fournies avec DB2 Connect version 8.2

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

Fonctions fournies avec DB2 Connect version 8.1 (notamment tous les FixPak 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
Fournit la fonction de détermination du programme d'application qui a émis une instruction particulière dans la mémoire cache pour instructions SQL dynamiques DB2 pour 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 lors de l'accès aux serveurs DB2 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.

Fonctions livrées avec DB2 Connect version 7.2

DB2 Connect version 7.2 inclut les améliorations suivantes :

- Support amélioré des technologies Microsoft Transaction Server (MTS) et COM+
- DB2 Connect Web Starter Kit
- DB2 Connect pour Linux sous S/390

Fonctions livrées avec DB2 Connect version 7.1

DB2 Connect version 7.1 inclut les améliorations suivantes :

- Concentrateur XA
- Améliorations apportées à la mise à jour multisite

Concepts connexes :

- «DB2 Connect», à la page 3

Référence connexe :

- «Bases de données hôte», à la page 6

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 :

OS/390 ou z/OS

DB2 UDB pour OS/390 et z/OS version 7 ou DB2 UDB pour z/OS version 8. Un sous-système DB2 Universal Database pour z/OS et OS/390 identifié par son LOCATION NAME (nom d'emplacement). 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 pour VSE fonctionnant sur une partition de base de données identifiée par son DBNAME

VM DB2 pour VM fonctionnant sur une machine virtuelle CMS identifiée par son DBNAME

OS/400

DB2 pour iSeries, un composant intégral du système d'exploitation OS/400. Une seule base de données peut exister sur un serveur iSeries, à moins que le serveur ne soit configuré pour utiliser des pools de stockage auxiliaires indépendants.

Concepts connexes :

- «DB2 Connect», à la page 3
- «DB2 Connect et instructions SQL», à la page 7

Référence connexe :

- «Utilitaires d'administration DB2 Connect», à la page 8
- «Support hôte et iSeries pour DB2 Connect» dans *Quick Beginnings for DB2 Connect Servers*

DB2 Connect et instructions SQL

DB2 Connect transfère des instructions SQL soumises par des programmes d'application à des serveurs de base de données hôte ou iSeries.

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
- 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 hôte ou iSeries. 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.

Divers serveurs de base de données hôte ou iSeries implémentent le SQL différemment. DB2 Connect prend entièrement en charge les implémentations IBM courantes, ainsi que DB2 pour OS/390 et z/OS, DB2 Server pour VSE & VM (anciennement SQL/DS) et DB2 pour iSeries. IBM SQL est fortement recommandé pour la gestion de l'indépendance des bases de données.

Concepts connexes :

- «DB2 Connect», à la page 3

Référence connexe :

- «Utilitaires d'administration DB2 Connect», à la page 8
- «Offres de produits DB2 Connect», à la page 3
- «Bases de données hôte», à la page 6

Utilitaires d'administration DB2 Connect

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

- L'Interpréteur de commandes (CLP) vous permet d'exécuter des instructions SQL dans un serveur de base de données hôte ou iSeries. 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).
- L'importation et l'exportation des utilitaires vous permet de charger, d'importer et d'exporter des données à partir d'un fichier ou vers un fichier situé sur un poste de travail ou un serveur de base de données hôte ou iSeries. 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 pour OS/390 ou 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 jobs correspondants fonctionnant sur un serveur de base de données hôte ou iSeries.

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.

Concepts connexes :

- «Moniteur du gestionnaire de bases de données» dans *System Monitor Guide and Reference*
- «Applications et utilitaires de liaison (DB2 Connect)», à la page 57
- «DB2 Connect», à la page 3
- «DB2 Connect et instructions SQL», à la page 7
- «Contrôle des performances à l'aide du moniteur de performances de Windows», à la page 74

WebSphere Federation Server et DB2 Connect

WebSphere Federation Server est une offre de produit distincte qui permet d'accéder aux données et d'intégrer des données via diverses sources de données multiconstructeur alors que DB2 Connect vous permet d'optimiser de grands volumes de données situés sur des hôtes et des serveurs de milieu de gamme existants.

WebSphere Federation Server vous aide à intégrer 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. WebSphere Federation Server fonctionne conjointement avec les serveurs DB2 Connect. WebSphere Federation Server fournit l'accès en lecture et en écriture natif à la famille de produits DB2, Informix, Oracle, Sybase, Teradata et aux bases de données Microsoft SQL Server. WebSphere Federation Server fournit un accès en lecture aux sources de données non relationnelles et de sciences biologiques, telles que BLAST, Documentum, Entrez, IBM Lotus Extended Search, les fichiers au format table et XML. Vous pouvez l'utiliser pour formuler des requêtes sur des données dans un système fédéré.

Concepts connexes :

- «DB2 Connect», à la page 3
- «Architecture de base de données relationnelle répartie (DRDA)», à la page 11

Référence connexe :

- «Offres de produits DB2 Connect», à la page 3

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

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.

Concepts connexes :

- «DB2 Connect et DRDA», à la page 12
- «Requêtes réparties», à la page 15
- «DRDA et accès aux données», à la page 11
- «Mises à jour multisites», à la page 61
- «Unité d'oeuvre éloignée», à la page 13

Tâches connexes :

- «Activation des mises à jour multisites à l'aide du Centre de contrôle», à la page 62

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 CLI) et OLE DB. Ces API peuvent être utilisées par les programmeurs afin de générer des applications en plusieurs langues.

Concepts connexes :

- «DB2 Connect et DRDA», à la page 12
- «Architecture de base de données relationnelle répartie (DRDA)», à la page 11

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 UDB pour iSeries, DB2 UDB pour OS/390 et z/OS, DB2 Server pour VSE & VM et d'autres serveurs de base de données 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, à la page 13 illustre le flot de données entre le poste de travail DB2 Connect et le serveur hôte ou iSeries dans l'éventualité de l'existence exclusive de clients locaux.

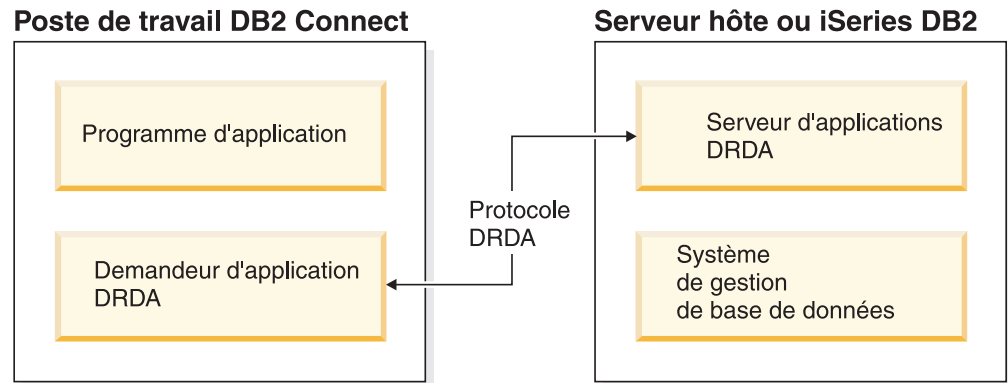


Figure 1. Flot de données entre un serveur DB2 Connect et un serveur hôte ou iSeries

Afin de mettre en oeuvre les connexions entre les systèmes de gestion de base de données serveur DRDA et les client de base de données, 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.

Concepts connexes :

- «Requêtes réparties», à la page 15
- «Architecture de base de données relationnelle répartie (DRDA)», à la page 11
- «Unité d'oeuvre éloignée», à la page 13

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).

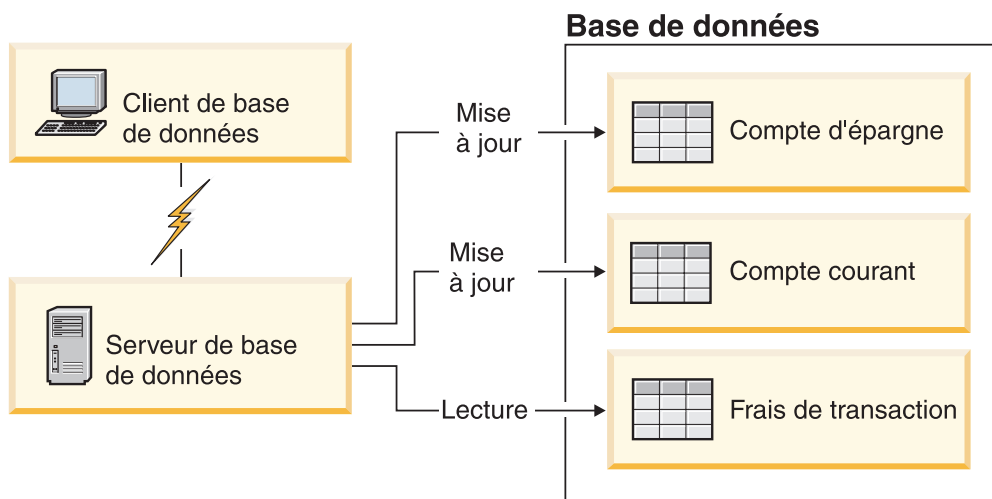


Figure 2. Utilisation d'une seule base de données au cours d'une transaction

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.

Concepts connexes :

- «DB2 Connect et DRDA», à la page 12
- «Requêtes réparties», à la page 15
- «Architecture de base de données relationnelle répartie (DRDA)», à la page 11
- «Unités d'oeuvre éloignées» dans *SQL Guide*

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 des tables de deux sous-systèmes DB2 pour OS/390 ou 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 Base de données DB2 pour Linux, UNIX et Windows, DB2 pour OS/390 et z/OS et DB2 UDB pour iSeries) et Oracle. Une prise en charge multiconstructeur est disponible lorsque vous utilisez DB2 Connect conjointement avec WebSphere 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 *alias de type nickname*) 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 alias de type nickname 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 alias de type nickname 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.

Concepts connexes :

- «DB2 Connect et DRDA», à la page 12
- «Architecture de base de données relationnelle répartie (DRDA)», à la page 11
- «Unité d'oeuvre éloignée», à la page 13

Chapitre 3. Scénarios DB2 Connect

Scénarios DB2 Connect

DB2 Connect peut fournir diverses solutions pour accéder à la base de données hôte ou iSeries. Cette rubrique élabore divers scénarios pouvant s'appliquer aux besoins spécifiques de votre environnement.

Concepts connexes :

- «DB2 Connect», à la page 3
- «DB2 Connect et serveurs d'applications», à la page 24
- «DB2 Connect et IBM WebSphere», à la page 21
- «DB2 Connect et moniteurs de traitement de transactions», à la page 27
- «DB2 Connect et applications Web», à la page 20
- «Produits serveur DB2 Connect en tant que serveurs de connectivité», à la page 19
- «Accès direct aux bases de données hôte», à la page 17

Scénarios

Accès direct aux bases de données hôte

La fonction de base de DB2 Connect offre une connexion directe vers une base de données hôte à partir des applications bureautiques fonctionnant sur les postes de travail Windows ou Linux. DB2 Connect Personal Edition constitue le procédé le plus simple pour fournir cette solution.

Chaque poste de travail sur lequel DB2 Connect Personal Edition est installé peut établir une connexion directe TCP/IP vers les serveurs DB2 UDB pour OS/390 et z/OS, DB2 UDB pour iSeries et Base de données DB2 pour Linux, UNIX et 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, à la page 18 illustre une connexion directe d'un serveur de base de données hôte ou iSeries vers un poste de travail sur lequel DB2 Connect Personal Edition est installé.

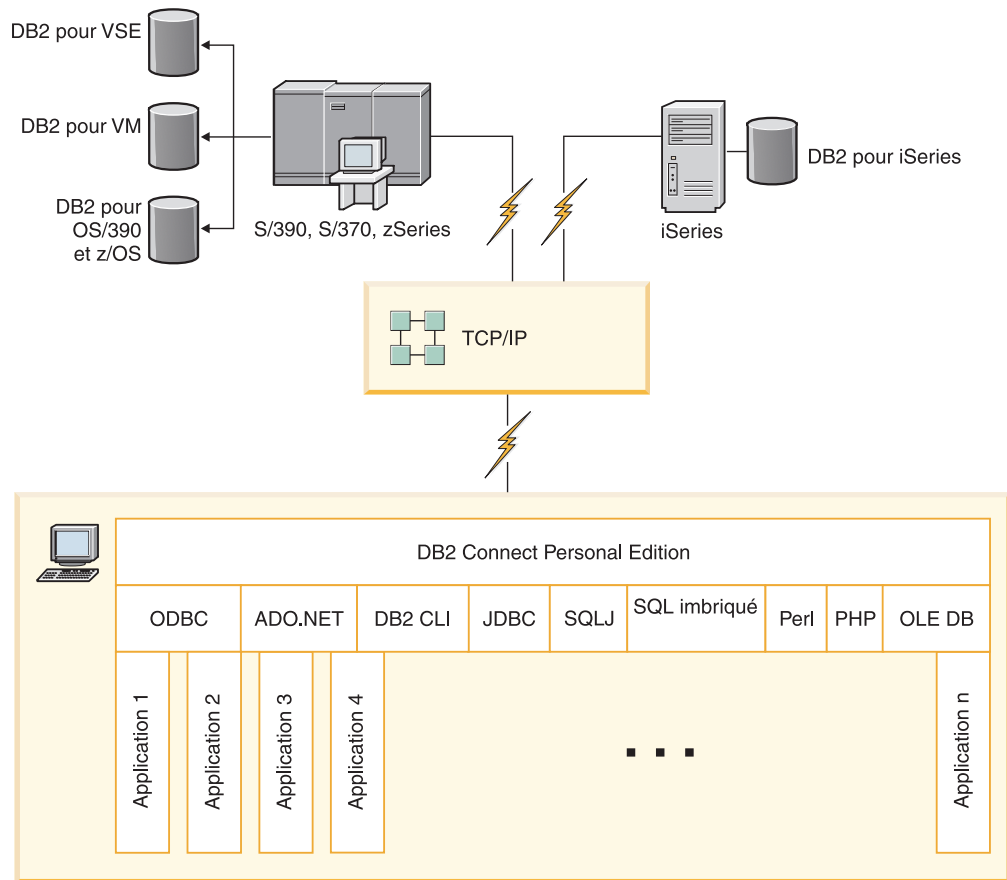


Figure 3. Connexion directe entre DB2 Connect et un serveur de base de données hôte ou iSeries

Remarques :

1. DB2 ne doit pas être nécessairement installé sur le poste de travail DB2 Connect. Si vous souhaitez un système de gestion de base de données relationnelle complet sur le poste de travail DB2 Connect, commandez DB2.
2. Le client DB2 fait désormais partie du module DB2 Connect et peut être installé si un client souhaite l'utiliser pour le développement d'application. En outre, DB2 Connect inclut désormais Stored Procedure Builder utilisé pour générer, tester et déployer des procédures mémorisées pour DB2 pour OS/390 et z/OS.
3. Les programmeurs en langage C qui développent des applications Windows utilisant la connectivité ODBC, OLE DB ou les objets données ActiveX Microsoft doivent utiliser le *kit de développement de logiciels Microsoft Open Database Connectivity*. Les programmeurs qui désirent développer des applications utilisant le langage de programmation Java peuvent utiliser n'importe quel environnement de développement Java.
4. Si une connexion au serveur de base de données DB2 pour z/OS activé Sysplex est perdue, le client tente de rétablir automatiquement la connexion.

Concepts connexes :

- «Accès aux données DB2 hôte ou iSeries via DB2 Connect Personal Edition» dans *Quick Beginnings for DB2 Connect Personal Edition*
- «DB2 Connect et serveurs d'applications», à la page 24
- «DB2 Connect et moniteurs de traitement de transactions», à la page 27

- «DB2 Connect et applications Web», à la page 20
- «Produits serveur DB2 Connect en tant que serveurs de connectivité», à la page 19
- «Scénarios DB2 Connect», à la page 17

Produits serveur DB2 Connect en tant que serveurs de connectivité

Un serveur DB2 Connect permet à plusieurs clients de se connecter à des données hôte ou iSeries et peut réduire de façon considérable l'effort requis pour établir et gérer l'accès aux données d'entreprise. La figure 4 illustre la solution IBM pour vos environnements dans lesquels vous souhaitez qu'un client DB2 crée une connexion indirecte vers un serveur de base de données hôte ou iSeries via un serveur DB2 Connect tel que DB2 Connect Enterprise Edition.

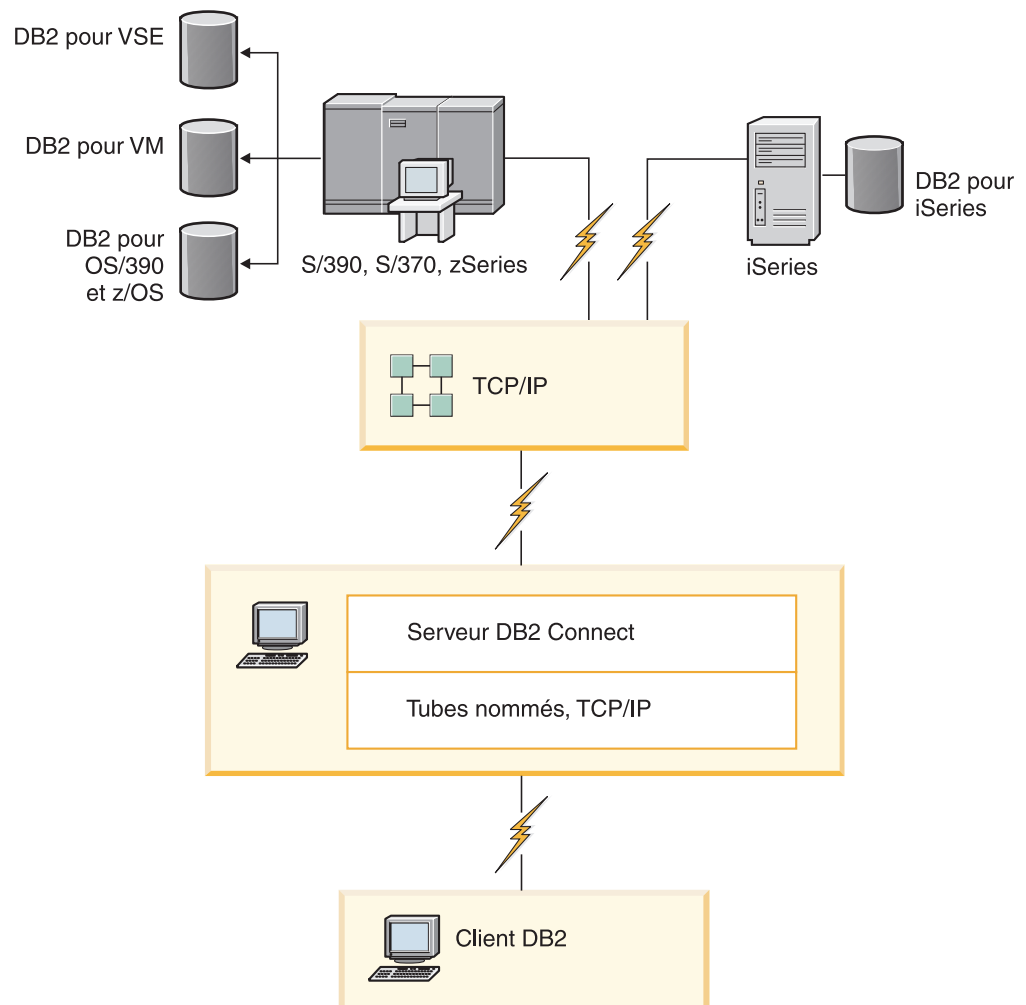


Figure 4. DB2 Connect Enterprise Edition

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.

Concepts connexes :

- «DB2 Connect», à la page 3
- «DB2 Connect et serveurs d'applications», à la page 24
- «DB2 Connect et moniteurs de traitement de transactions», à la page 27
- «DB2 Connect et applications Web», à la page 20
- «Scénarios DB2 Connect», à la page 17

Référence connexe :

- «Offres de produits DB2 Connect», à la page 3

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 rédigés en langage C++ ou dans un langage de script tel que Perl.

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. En outre, elle connaît également des problèmes d'évolutivité qui affectent les opérations de commerce électronique à grande échelle. Chaque fois qu'une application CGI est invoquée, un nouveau processus est créé sur le serveur Web. Chaque instance doit établir sa propre connexion à la base de données et chaque

instance soumet 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.

Concepts connexes :

- «DB2 Connect et serveurs d'applications», à la page 24
- «DB2 Connect et IBM WebSphere», à la page 21
- «DB2 Connect et moniteurs de traitement de transactions», à la page 27
- «Produits serveur DB2 Connect en tant que serveurs de connectivité», à la page 19
- «DB2 Connect sur le serveur Web», à la page 23

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, OS/390, z/OS, OS/400, Solaris operating system, 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. Si DB2 UDB pour OS/390 et z/OS, DB2 pour VM, DB2 pour VSE, et DB2 UDB pour iSeries, WebSphere sont déployés 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/>

Concepts connexes :

- «Scénarios DB2 Connect», à la page 17

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 les applets et les applications qui vous permettent d'utiliser Java dans toutes les étapes de la transaction Web. Les solutions IBM vous offre un mélange de techniques, ce qui signifie que vous pouvez utiliser les solutions de script telles que Perl DBI ou Microsoft Active Server Pages avec DB2, ou migrer vers une mise en oeuvre plus robuste fournie par un serveur d'applications Java tel qu'IBM WebSphere.

Il existe deux interfaces de programmation (API) pour les programmeurs Java. La première, l'interface JDBC, est prise en charge pour une utilisation de Java en vue de développer des applets de données Java, des applications Java ainsi que des servlets Java, le langage de script Java server pages (JSP) et les Enterprise Java Beans (EJB). JDBC est une API d'appel de méthode ou de niveau d'appel. L'autre API Java est SQLJ. Le SQLJ vous permet de spécifier le SQL en ligne dans un programme Java. DB2 peut utiliser les deux API, du côté serveur ou client de la transaction Web.

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

Pour DB2 pour OS/390 et z/OS, DB2 pour VSE et VM, et DB2 UDB pour iSeries, il existe deux procédés de déploiement pour une application Java. Vous pouvez utiliser la connectivité directe fournie par DB2 Connect Personal Edition avec le protocole TCP/IP, ou vous pouvez choisir d'utiliser le serveur DB2 Connect qui vous offrira la connectivité au serveur de base de données hôte ou iSeries.

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 DB2 doit être installé sur le serveur Web.

Pour DB2 pour OS/390 et z/OS, le composant clé est un serveur DB2 Connect fonctionnant sur un serveur de niveau intermédiaire. Ce composant offre l'activation du serveur JDBC en plus de la connexion au serveur DB2 pour OS/390 et z/OS, DB2 pour VSE et VM, ou DB2 UDB pour iSeries. Une fois encore, le client n'a pas besoin de posséder d'autre logiciel spécifique que le navigateur Web.

IBM offre une prise en charge étendue et les outils de développement des applications et applets Java. Pour le développement d'applications de base de données, DB2 Database Enterprise Developer Edition fournit Rational Web Developer, DB2 Developer Workbench, DB2 Embedded Application Server, Cloudscape version 10.2 ainsi que DB2 et DB2 Connect pour effectuer des tests. Les outils tiers, tels que NetBeans, Borland JBuilder ou Symantec Visual Cafe fonctionnent également avec les solutions de base de données IBM.

Concepts connexes :

- «DB2 Connect sur le serveur Web», à la page 23
- «Scénarios DB2 Connect», à la page 17

Référence connexe :

- «Offres de produits DB2 Connect», à la page 3

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 fonctionnant sur les systèmes zSeries, iSeries, VM et VSE, un serveur DB2 Connect est nécessaire sur le serveur Web. Les serveurs DB2 Connect vous offrent les bibliothèques et les interfaces de communication pour activer l'accès des serveurs Web aux plateformes hôte ou iSeries. Le protocole TCP/IP peut être utilisé pour les communications entre le serveur Web et une base de données fonctionnant sur zSeries, iSeries, VM ou VSE.

Remarque : Les solutions IBM vous permettent de travailler avec plusieurs bases de données au sein d'un même script CGI ou de 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 coeur 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. Base de données DB2 pour Linux, UNIX et Windows, et DB2 UDB sous OS/390 et z/OS, DB2 UDB pour iSeries et DB2 pour VSE partagent la même convention de paramètre pour appeler les procédures mémorisées.

Comme pour l'interface CGI, le navigateur Web soumet le formulaire au serveur Web sur lequel le script CGI 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 qui affluent entre le script CGI 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.

Concepts connexes :

- «Scénarios DB2 Connect», à la page 17

Référence connexe :

- «Offres de produits DB2 Connect», à la page 3

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 de type DB2 UDB pour OS/390 et z/OS, DB2 UDB pour iSeries ou DB2 pour VM et 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'applications 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 mettre des données à jour dans une base de données DB2 pour OS/390 et z/OS et Base de données DB2 pour Linux, UNIX et Windows au cours de la même transaction. 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.

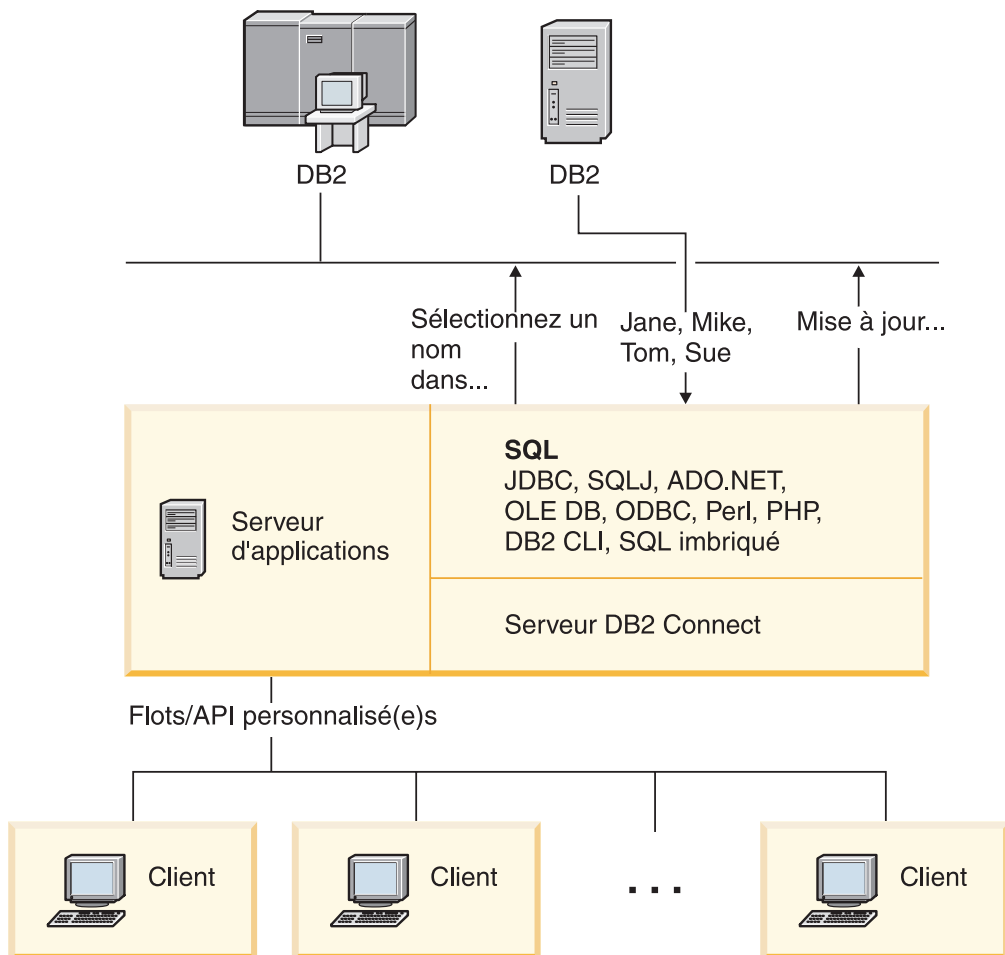


Figure 5. DB2 Connect prend en charge les serveurs d'applications

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.

Concepts connexes :

- «Concentrateur de connexion», à la page 98
- «Regroupement de connexions», à la page 95
- «DB2 Connect», à la page 3
- «DB2 Connect et moniteurs de traitement de transactions», à la page 27
- «DB2 Connect et applications Web», à la page 20

Référence connexe :

- «Offres de produits DB2 Connect», à la page 3
- «Considérations de sécurité DB2 Connect applicables à DB2 pour OS/390 et z/OS», à la page 53

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 fonctionnant sous Linux, UNIX et Windows d'exécuter des transactions dans le réseau LAN distant, les serveurs de base de données hôte ou iSeries et de coordonner ces transactions à l'aide d'un moniteur TP.

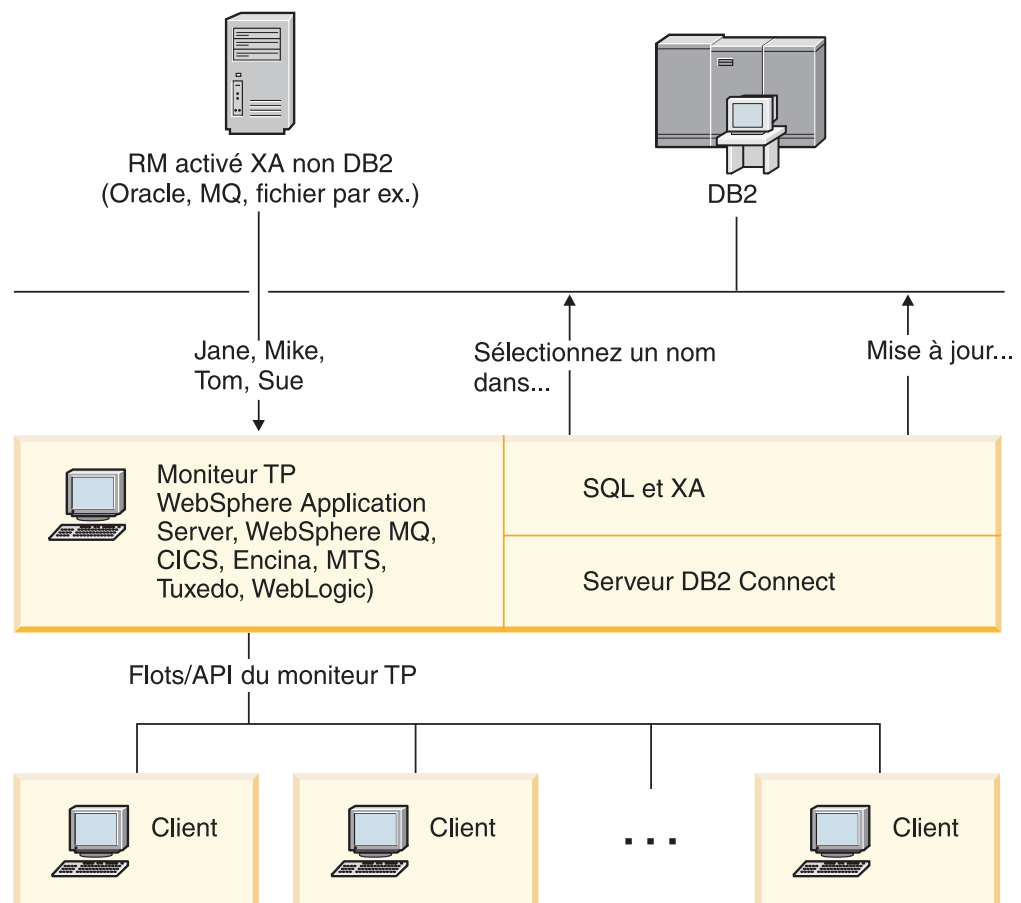


Figure 6. Prise en charge de DB2 Connect pour les moniteurs TP

Dans figure 6, à la page 28, 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 iSeries, zSeries distants et LAN peuvent être utilisés au sein 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, la première base de données est une base de données DB2 Universal Database pour OS/390 et z/OS et la seconde, 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 bases de données iSeries et LAN DB2, auxquels un utilisateur accède via DB2 Connect, sont des gestionnaires de ressources compatibles XA. Par conséquent, tout moniteur TP possédant un gestionnaire de transactions compatible XA peut utiliser des bases de données hôte ou iSeries et DB2 LAN au sein des applications métier lors de l'exécution de transactions.

Concepts connexes :

- «Considérations sur la configuration des gestionnaires de transactions XA» dans *Administration Guide: Planning*
- «Considérations sur la sécurité pour les gestionnaires de transactions XA» dans *Administration Guide: Planning*
- «Modèle de traitement réparti des transactions (DTP) X/Open» dans *Administration Guide: Planning*

- «Fonction XA prise en charge pour DB2 Database pour Linux, UNIX et Windows» dans *Administration Guide: Planning*

Tâches connexes :

- «Configuration de DB2 Connect avec un gestionnaire de transactions compatible avec XA», à la page 64
- «Mise à jour des serveurs de base de données hôte ou iSeries avec un gestionnaire de transactions compatible XA» dans *Administration Guide: Planning*

Partie 2. Référence

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

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 hôte ou iSeries auquel DB2 Connect accède.
- Le *répertoire des services de connexion de la base de données* qui contient les informations spécifiques aux bases de données du serveur de base de données hôte ou iSeries.

Remarques :

1. Avant de mettre à jour ces répertoires, vous devez configurer les communications sur le serveur de base de données hôte ou iSeries 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).

Procédure :

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. Mise à jour des informations relatives aux serveurs de base de données éloignés dans les répertoires

Tâches connexes :

- «Mise à jour des répertoires avec les informations sur les machines serveur de base de données éloignées» dans *Administration Guide: Implementation*

Référence connexe :

- «Commande LIST DATABASE DIRECTORY» dans *Command Reference*
- «Commande LIST DCS DIRECTORY» dans *Command Reference*
- «Commande LIST NODE DIRECTORY» dans *Command Reference*
- «Feuille de travail de personnalisation du répertoire», à la page 40

Valeurs du répertoire système des bases 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 hôte ou iSeries. 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, DCE, KERBEROS et DATA_ENCRYPT.

Concepts connexes :

- «Valeurs du répertoire des noeuds», à la page 34
- «Mise à jour des répertoires de bases de données», à la page 33

Valeurs du répertoire des noeuds

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

Nom de noeud

Alias du système du serveur de base de données hôte ou iSeries sur lequel la base de données éloignée se trouve. 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 éloignés DB2 pour OS/390 et z/OS, le nom d'hôte apparaît dans le message DSNL004I (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 éloignés DB2 pour OS/390 et z/OS, le numéro de port est défini dans l'ensemble de données d'amorce (BSDS) en tant que PORT et est également indiqué dans le message 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 Universal Database pour z/OS et OS/390 attribue un numéro de port (RESPORT) à utiliser pour la resynchronisation des connexions entrantes dans DB2 Universal Database pour z/OS et OS/390 uniquement. Aucun nom de service ne doit être défini.

Concepts connexes :

- «Types de sécurité pris en charge avec DB2 Connect», à la page 54
- «Mise à jour des répertoires de bases de données», à la page 33

Valeurs du répertoire DCS

Vous pouvez spécifier les informations suivantes dans le répertoire DCS :

Nom de la base de données

Alias défini par l'utilisateur pour le serveur de base de données hôte ou iSeries. 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, 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 la base de données cible

La base de données sur le système du serveur de base de données hôte ou iSeries :

OS/390 et z/OS

Un sous-système DB2 Universal Database pour z/OS et OS/390 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)

OS/400 et z/OS

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 que l'application se déconnectera de la base de données du serveur de base de données hôte ou iSeries lorsque l'un des codes SQLCODE 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 *Guide des messages*.

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 est configuré dans le répertoire DCS sur le poste de travail DB2 Connect et qu'une application client émet une demande d'interruption alors qu'il est connecté au serveur de base de données hôte ou iSeries, DB2 Connect procède à l'interruption en supprimant la connexion et en annulant l'unité d'oeuvre. 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 hôte ou iSeries 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 Universal Database pour z/OS et OS/390 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.

Une nouvelle variable de profil (environnement ou registre) a également été introduite, la variable DB2SYSPLEX_SERVER, et peut être utilisée pour désactiver la prise en charge SYSPLEX au niveau du poste de travail.

,,,,,LOCALDATE="<value>"

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. 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.

Concepts connexes :

- «Mise à jour des répertoires de bases de données», à la page 33

Référence connexe :

- «Feuille de travail de personnalisation du répertoire», à la page 40

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.

Valeurs 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)	

Remarques :

1. Le numéro de port TCP/IP par défaut pour DRDA est 446
2. Sauf si vous savez si le serveur de base de données hôte ou iSeries 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=\"\"YYMMDD\"\"\""	

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	

Concepts connexes :

- «Valeurs du répertoire DCS», à la page 35
- «Valeurs du répertoire des noeuds», à la page 34
- «Valeurs du répertoire système des bases de données», à la page 33
- «Mise à jour des répertoires de bases de données», à la page 33

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 dont le port est défini sur l'hôte ou le serveur de base de données iSeries tout en acceptant le mappage par défaut pour les applications développées pour l'environnement client/serveur. Effectuez également les étapes suivantes :

- 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.

Concepts connexes :

- «Mise à jour des répertoires de bases de données», à la page 33

Référence connexe :

- «Feuille de travail de personnalisation du répertoire», à la page 40

Traitement des données bidirectionnelles (BiDi)

La section suivante concerne uniquement les serveurs OS/390 et z/OS. Cette fonction ne doit pas être activée pour un serveur DB2 pour iSeries car la prise en charge intégrale des données bidirectionnelles est déjà définie.

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 sous OS/390 ou z/OS les données sont généralement au format SHAPED VISUAL. Par conséquent, sans prise en charge des attributs BiDi, les données envoyées de DB2 pour OS/390 et z/OS vers 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 hébreu DB2 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 DB2. 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) vers le CCSID 62213 (chaîne BiDi de type 5).

Remarques :

1. La variable d'environnement ou la valeur de registre DB2BIDI doit être définie sur YES pour que le paramètre BIDI prenne effet.
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="YMMDD",,
```

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=\"\"YMMDD\"\""
```

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="YMMDD"
Commentaire =
Niveau d'édition du répertoire DCS = 0x0100
```

Concepts connexes :

- «Support bidirectionnel avec DB2 Connect» dans *Administration Guide: Planning*

Tâches connexes :

- «Activation du support bidirectionnel» dans *Administration Guide: Planning*

Référence connexe :

- «CCSID spécifiques au traitement des données bidirectionnelles» dans *Administration Guide: Planning*

Chapitre 5. Sécurité

Considérations sur l'authentification DB2

En tant qu'administrateur DB2 Connect, avec votre administrateur de base de données hôte ou iSeries, vous pouvez déterminer l'emplacement de validation des noms d'utilisateur et des mots de passe :

- Au niveau du client
- Au niveau du serveur hôte ou iSeries
- 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.

Depuis DB2 Connect Version 8.2.2 (équivalent à la version 8.1, FixPak 9), la passerelle n'est plus un composant passif lors de la négociation d'authentification. Au contraire, elle prend un rôle actif. Le type d'authentification indiqué dans l'entrée de répertoire de base de données de 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 spécifié dans l'entrée de répertoire de base de données, l'authentification SERVER sera le type de serveur demandé par défaut. Toutefois, la négociation aura toujours lieu entre le client et le serveur si le serveur ne prend pas en charge l'authentification SERVER. Ce comportement présente un contraste avec le client qui est défini par défaut sur SERVER_ENCRYPT si aucun type d'authentification n'a été spécifié.

Le type d'authentification catalogué sur la passerelle n'est pas utilisé si DB2NODE ou l'option SQL_CONNECT_NODE de l'interface client API a été définie sur le client. Dans ces cas là, la négociation s'effectue toujours strictement entre le client et le serveur.

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

CLIENT

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

SERVER

Le nom d'utilisateur et le mot de passe sont validés par la base de données du serveur hôte ou iSeries.

SERVER_ENCRYPT

Pour l'authentification SERVER, le nom d'utilisateur et le mot de passe sont validés par le serveur de la base de données hôte ou iSeries mais les mots de passe transférés sont chiffrés par le client.

DATA_ENCRYPT

Offre la possibilité de chiffrer des données utilisateur lors de communications client/serveur.

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.

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.

Concepts connexes :

- «Types de sécurité pris en charge avec DB2 Connect», à la page 54

Référence connexe :

- «Astuces et conseils supplémentaires pour la sécurité OS/390 et z/OS», à la page 53
- «Considérations de sécurité DB2 Connect applicables à DB2 pour OS/390 et z/OS», à la page 53

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 : la relation entre le KDC qui contrôle le poste de travail client et le système iSeries, OS/390 ou z/OS. 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 des clients à DB2 pour OS/390 et 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 DB2 :

Version 8

DB2 Connect:

Version 8

DB2 UDB pour OS/390 et z/OS :

Version 7

Concepts connexes :

- «Types de sécurité pris en charge avec DB2 Connect», à la page 54

Référence connexe :

- «Considérations de sécurité DB2 Connect applicables à DB2 pour OS/390 et z/OS», à la page 53

Connexions sécurisées

Connexions sécurisées via DB2 Connect

Certains serveurs de base de données DB2 prennent en charge les contextes sécurisés. Un *contexte sécurisé* permet à l'administrateur de base de données, entre autres, de définir les conditions sous lesquelles l'application client sera autorisée à 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, une connexion implicite et une connexion explicite. Lors de la création d'une connexion, l'obtention d'une connexion sécurisée explicite, d'une connexion sécurisée implicite ou d'une connexion classique dépend de votre demande, connexion sécurisée ou non, ou dépend des critères définis dans le contexte sécurisé sur le serveur, comme récapitulé dans tableau 4.

Tableau 4. Type de connexions obtenu par les différentes combinaisons d'actions

	La connexion réponds aux critères du serveur pour être sécurisée	La connexion ne réponds pas aux critères du serveur pour être sécurisée
Vous avez demandé une connexion sécurisée	Connexion sécurisée explicite	Connexion classique et renvoi d'un avertissement SQL20360W (SQLSTATE 01679).
Vous n'avez pas demandé de connexion sécurisée	Connexion sécurisée implicite	Connexion classique

Une *connexion sécurisée implicite* est identique à une connexion classique excepté qu'elle octroie des privilèges temporaires à l'utilisateur pendant la connexion. Les privilèges octroyés (le cas échéant) sont spécifiés dans le contexte sécurisé qui a engendré la connexion sécurisée.

Les connexions sécurisées implicites sont créées par n'importe quelle application qui se connecte à l'aide de DB2 Connect. Elles sont créées et utilisées de la même

manière que les connexions classiques. Cela signifie qu'un changement de code n'est nécessaire pour qu'une application existante profite des connexions sécurisées implicites pourvu que l'application se connecte via DB2 Connect.

Une *connexion sécurisée explicite* octroie des privilèges temporaires à l'utilisateur de la même manière qu'une connexion sécurisée implicite. En outre, une connexion sécurisée explicite vous permet de modifier l'ID autorisation utilisé pendant l'exécution d'actions dans cette connexion. La modification de l'ID autorisation sur une connexion sécurisée explicite se rapporte à un *transfert d'utilisateurs*. Les ID autorisation vers lesquels vous pouvez basculer ainsi que la nécessité pour l'ID autorisation d'avoir un mot de passe pour le transfert sont définis dans le cadre du contexte sécurisé qui autorise la création de la connexion sécurisée.

Le transfert d'utilisateur peut réduire considérablement le temps système de partage d'une connexion entre plusieurs utilisateurs, en particulier pour les noms d'utilisateur qui ne nécessitent pas de mot de passe car dans ce cas, le serveur de base de données n'authentifie pas l'ID autorisation. Lors de l'utilisation de la fonction, toutefois, vous devez vous assurer que votre application n'autorise pas le transfert vers un ID autorisation sans validation ni authentification de cet ID autorisation. Sinon, vous créez une faille dans la sécurité de votre système.

Des connexions sécurisées explicites peuvent être créées et le changement d'utilisateur peut avoir lieu lors de la connexion via DB2 Connect à l'aide de CLI ou JDBC, y compris des connexions établies au moyen de XA. La création d'une connexion sécurisée explicite et le changement d'utilisateur nécessitent de configurer des attributs de connexions particuliers. Cela signifie que les applications existantes devront être modifiées afin qu'elles puissent bénéficier des connexions sécurisées explicites.

Outre les différences mentionnées ci-dessus, vous pouvez utiliser une connexion sécurisée (implicite ou explicite) de la même manière qu'une connexion classique. Toutefois, vous devez prendre soin de vous déconnecter explicitement d'une connexion sécurisée explicite dès que vous avez terminé, même si son état indique qu'elle est arrêtée ou déconnectée. Sinon, les ressources utilisées par la connexion risquent de ne pas être libérées. Il ne s'agit pas d'un problème lié aux connexions sécurisées implicites.

Remarques :

1.

Important : Le changement d'utilisateurs sans saisie de mot de passe ignore l'authentification du serveur de la base. Votre application ne doit pas autoriser de changement sans mot de passe sauf si cette application a déjà validé et authentifié cet ID autorisation. Sinon, vous créez une faille dans la sécurité.

2. L'authentification CLIENT n'est pas recommandée avec les connexions sécurisées explicites. Cette recommandation ne s'applique pas aux connexions sécurisées implicites.
3. Les applications utilisant des connexions sécurisées explicites doivent être exécutées sur des postes sécurisés, protégés par un mot de passe et accessibles uniquement par du personnel autorisé. Cette recommandation ne s'applique pas aux connexions sécurisées implicites.

Concepts connexes :

- «Support contextuel sécurisé IBM DB2 Driver pour JDBC et SQLJ» dans *Developing Java Applications*

Tâches connexes :

- «Création et arrêt d'une connexion sécurisée via CLI», à la page 49
- «Changement d'utilisateur sur une connexion sécurisée via CLI», à la page 50

Création et arrêt d'une connexion sécurisée via CLI

Si le serveur de base de données auquel vous vous connectez l'autorise, 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 en utilisez un, vous devez vérifier que le gestionnaire de transactions est configuré pour indiquer la valeur de configuration TCTX sur TRUE quand il appelle xa_open. Si tel est le cas, toutes les connexions qui le peuvent seront des connexions sécurisées explicites. Pour vérifier qu'une connexion est une connexion sécurisée explicite, allez à l'étape 3.

Conditions préalables :

- 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 spécifié dans le contexte sécurisé. L'ID autorisation système d'une connexion sécurisée est l'ID autorisation que vous avez fourni au serveur comme nom d'utilisateur lors de la création de la connexion. Pour sécuriser votre connexion dans un contexte sécurisé particulier, l'ID autorisation système doit être celui spécifié dans ce contexte sécurisé. Demandez à votre administrateur sécurité un ID autorisation système valide ainsi que le mot de passe correspondant.

Procédure :

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 à une base de données classique, en appelant la fonction SQLConnect par exemple. Utilisez l'ID autorisation système comme nom d'utilisateur et le mot de passe correspondant. Veillez à vérifier les erreurs et avertissements, notamment celles et ceux répertoriés dans la table tableau 5, à la page 50.

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, la connexion est établie et correspond à une connexion sécurisée explicite.

- (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`. Si l'attribut est défini sur `SQL_TRUE`, la connexion est une connexion sécurisée explicite.
- Dès que vous avez terminé avec votre connexion, vous devez prendre soin de vous déconnecter explicitement, même si l'état de la connexion indique qu'elle est arrêtée ou déconnectée. Si vous ne vous déconnectez pas explicitement d'une connexion sécurisée explicite, certaines ressources utilisées par la connexion risquent de ne pas être libérées.

Remarques :

- L'authentification `CLIENT` n'est pas recommandée avec les connexions sécurisées explicites. Cette recommandation ne s'applique pas aux connexions sécurisées implicites.
- Les applications utilisant des connexions sécurisées explicites doivent être exécutées uniquement sur des postes sécurisés, protégés par un mot de passe et accessibles uniquement par du personnel autorisé. Cette recommandation ne s'applique pas aux connexions sécurisées implicites.

Concepts connexes :

- «Connexions sécurisées via DB2 Connect», à la page 47

Tâches connexes :

- «Changement d'utilisateur sur une connexion sécurisée via CLI», à la page 50

Référence connexe :

- «Liste des attributs de connexion (CLI)» dans *Call Level Interface Guide and Reference, Volume 2*

Changement d'utilisateur sur une connexion sécurisée via CLI

Vous pouvez changer d'utilisateur sur une connexion sécurisée explicite via CLI. Pour une description plus détaillée sur ce que le changement d'utilisateur signifie, consultez la rubrique Connexions sécurisées via DB2 Connect à la section Notions associées.

Conditions préalables :

- La connexion doit avoir été correctement créée comme connexion sécurisée explicite.
- La connexion sécurisée explicite ne doit pas se trouver dans une transaction.
- Le contexte sécurisé qui permet la création de la connexion sécurisée explicite doit être configuré pour autoriser le changement d'ID autorisation.

Procédure :

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

1. Appelez la fonction `SQLSetConnectAttr` pour définir l’attribut `SQL_ATTR_TRUSTED_CONTEXT_USERID`. Définissez-le sur l’ID autorisation vers lequel vous souhaitez basculer.

```
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 la table 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’a pas abouti.
CLI0197E	La connexion n’est pas une connexion sécurisée.
CLI0124E	Un incident est survenu à cause de la valeur fournie. Vérifiez qu’il ne s’agit pas d’une valeur NULL, trop longue, etc.
CLI0196E	La connexion est impliquée dans une unité de travail qui l’empêche de changer d’utilisateurs. Pour pouvoir changer d’utilisateur, la connexion ne doit pas se trouver dans une transaction.

2. (Facultatif, sauf si le contexte sécurisé qui a autorisé cette connexion sécurisée nécessite un mot de passe pour l’ID autorisation vers lequel vous basculez) Appelez la fonction `SQLSetConnectAttr` pour définir l’attribut `SQL_ATTR_TRUSTED_CONTEXT_PASSWORD`. Définissez-le sur le mot de passe du nouvel ID autorisation.

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

Vérifiez les erreurs et avertissements, ceux répertoriés dans la table tableau 6 et dans la table tableau 7.

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

SQLCODE	Signification
CLI0198E	L’attribut <code>SQL_ATTR_TRUSTED_CONTEXT_USERID</code> n’a pas encore été défini.

3. Poursuivez comme avec une connexion classique. Si vous utilisez un gestionnaire de transaction XA, une tentative de changement d’utilisateur est effectuée lors de la requête suivante, sinon elle s’effectue juste avant le déclenchement de l’appel de fonction suivant qui permet d’accéder à la base de données (`SQLExecDirect` par exemple). Dans les deux cas, outre la vérification des erreurs et avertissements, vérifiez également les erreurs répertoriées dans tableau 8, à la page 52. Cette liste d’erreurs indique que le changement d’utilisateur a échoué.

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

SQLCODE	Signification
SQL1046N	Le contexte sécurisé qui permet cette connexion sécurisée n'est pas configuré pour autoriser le changement d'ID autorisation vers lequel vous tentez de basculer. Vous devez modifier le contexte sécurisé pour pouvoir basculer vers cet ID autorisation.
SQL30082N	Le mot de passe fourni est incorrect pour l'ID autorisation vers lequel vous basculez.
SQL0969N avec une erreur native -20361	Certaines contraintes de niveau de base de données vous empêchent de basculer vers cet utilisateur.

En cas d'échec du changement d'utilisateur, l'état de la connexion indiquera 'non connecté' jusqu'à ce que vous réussissiez à basculer vers un autre utilisateur. Vous pouvez changer d'utilisateurs sur une connexion sécurisée en état 'non connecté' mais vous ne pouvez pas accéder au serveur de base de données. Une connexion en état 'non connecté' conserve son état jusqu'à ce que vous réussissiez à changer d'utilisateurs.

Remarques :

1.

Important : Le changement d'utilisateurs sans saisie de mot de passe ignore l'authentification du serveur de la base. Votre application ne doit pas autoriser de changement sans mot de passe sauf si cette application a déjà validé et authentifié cet ID autorisation. Sinon, vous créez une faille dans la sécurité.

2. Une valeur NULL définie pour l'attribut SQL_ATTR_TRUSTED_CONTEXT_USERID revient à spécifier l'ID autorisation du système de contexte sécurisé (l'ID utilisateur qui a servi à créer la connexion sécurisée explicite).
3. La connexion est immédiatement réinitialisée dès que vous définissez la bonne valeur de l'attribut de connexion SQL_ATTR_TRUSTED_CONTEXT_USERID sur une connexion sécurisée explicite. Le résultat de cette action est identique à la création d'une nouvelle connexion à l'aide des attributs de connexion d'origine de cette connexion. Cette réinitialisation a lieu même si la valeur que vous avez définie pour l'attribut de connexion correspond à l'ID autorisation système ou une valeur NULL ou la valeur que l'attribut conserve en attente.
4. Si l'attribut SQL_ATTR_TRUSTED_CONTEXT_PASSWORD est défini, le mot de passe sera authentifié lors du processus de changement d'utilisateurs, même si le contexte sécurisé qui autorise la connexion sécurisée ne nécessite pas d'authentification pour cet ID autorisation lors d'un changement d'utilisateurs. Il en résulte une surcharge inutile. Cette règle ne s'applique pas à l'ID autorisation système du contexte sécurisé. Si cet ID autorisation système du contexte sécurisé ne nécessite pas d'authentification lors du changement vers cet utilisateur, alors il n'est pas authentifié même si un mot de passe est fourni.

Concepts connexes :

- «Connexions sécurisées via DB2 Connect», à la page 47

Tâches connexes :

- «Création et arrêt d'une connexion sécurisée via CLI», à la page 49

Référence connexe :

- «Liste des attributs de connexion (CLI)» dans *Call Level Interface Guide and Reference, Volume 2*
- «Fonction SQLSetConnectAttr (CLI) - Définition des attributs de connexion» dans *Call Level Interface Guide and Reference, Volume 2*

Considérations de sécurité DB2 Connect applicables à DB2 pour OS/390 et z/OS

Cette rubrique décrit les remarques relatives à la sécurité DB2 Connect, notamment les types d'authentification et les paramètres de sécurité. Elle fournit également des conseils supplémentaires pour la sécurité aux utilisateurs de DB2 pour OS/390 et z/OS.

Concepts connexes :

- «Considérations sur l'authentification DB2», à la page 45
- «Types de sécurité pris en charge avec DB2 Connect», à la page 54

Référence connexe :

- «Astuces et conseils supplémentaires pour la sécurité OS/390 et z/OS», à la page 53

Astuces et conseils supplémentaires pour la sécurité OS/390 et z/OS

Cette rubrique fournit des conseils supplémentaires sur la sécurité lorsque DB2 Connect se connecte à un serveur de base de données DB2 pour OS/390 et z/OS.

Zone de sécurité étendue :

Vérifiez que la zone de sécurité étendue DB2 OS/390 et z/OS est définie sur YES. Cette zone apparaît dans le panneau DB2 pour OS/390 et z/OS DSNTIPR.

Codes de sécurité étendue :

Jusque DB2 Universal Database pour z/OS et OS/390 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 Universal Database pour z/OS et OS/390 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 Universal Database pour z/OS et OS/390 ZPARM relatif à la sécurité étendue doit être défini sur YES. Utilisez le panneau d'installation DB2 Universal Database pour z/OS et OS/390 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 :

Si vous souhaitez fournir une prise en charge de l'option de sécurité DB2AUTHENTICATION=CLIENT , utilisez le panneau d'installation DB2 Universal Database pour z/OS et OS/390 DSNTIP4 (panneau DDF 2) pour définir l'option Sécurité TCP/IP déjà vérifiée sur YES.

Sécurité des applications Java et ODBC de bureau :

Les applications ODBC de poste de travail et Java utilisent le SQL dynamique. Cela peut engendrer des problèmes de sécurité sur certaines installations. DB2 Universal Database pour z/OS et OS/390 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 Universal Database pour z/OS et OS/390 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 permet également de modifier le mot de passe.

Référence connexe :

- «Considérations de sécurité DB2 Connect applicables à DB2 pour OS/390 et z/OS», à la page 53
- «Commande BIND» dans *Command Reference*
- «CONNECT (Type 1) (instruction)» dans *SQL Reference, Volume 2*

Types de sécurité 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 de sécurité pour les connexions TCP/IP

Le protocole de communication TCP/IP ne prend pas en charge les options de sécurité 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 de sécurité valides

Scénario	Paramètre d'authentification	Validation
1	CLIENT	Client
2	SERVER	Serveur de base de données hôte ou iSeries
3	SERVER_ENCRYPT	Serveur de base de données hôte ou iSeries
4	KERBEROS	sécurité Kerberos
5	DATA_ENCRYPT	Serveur de base de données hôte ou iSeries

Rubrique relative aux types de sécurité

La rubrique suivante concerne les connexions décrites ci-dessus et répertoriées dans la section tableau 9. Chaque scénario est décrit en détails, 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 hôte ou iSeries. L'ID utilisateur et le mot de passe sont envoyés à travers le réseau, du client distant vers le serveur DB2 Connect et du serveur DB2 Connect vers le serveur de base de données hôte ou iSeries.
- 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.

Concepts connexes :

- «Considérations sur l'authentification DB2», à la page 45

Référence connexe :

- «Astuces et conseils supplémentaires pour la sécurité OS/390 et z/OS», à la page 53
- «Considérations de sécurité DB2 Connect applicables à DB2 pour OS/390 et z/OS», à la page 53

Chapitre 6. Utilitaires et applications de liaison

Applications et utilitaires de liaison (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 un processus de traitement des fichiers de liens par un serveur de base de données hôte ou iSeries.

Comme plusieurs utilitaires fournis avec DB2 Connect sont développés avec le SQL imbriqué, ils doivent être liés à un serveur de base de données hôte ou iSeries avant de pouvoir être utilisés avec ce système. Si vous n'utilisez pas les utilitaires et interfaces de DB2 Connect, vous ne devez pas les lier à chacun de vos serveurs de base de données hôte ou iSeries. Les listes de fichiers de liens requis par ces utilitaires se trouvent dans les fichiers suivants :

- ddcsmvs.lst pour OS/390 ou z/OS
- ddcsvse.lst pour VSE
- ddcsvm.lst pour VM
- ddcs400.lst pour OS/400

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 tous les serveurs de base de données hôte ou iSeries avant de pouvoir être utilisés 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 UDB pour OS/390 et z/OS via DB2 Connect Enterprise Edition sur un serveur Windows, procédez comme suit :

- 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 hôte ou iSeries à partir duquel vous procédez à la liaison, procédez comme suit :

1. Veillez à posséder les droits d'accès suffisants pour votre système de gestion du serveur de base de données hôte ou iSeries :

OS/390 ou z/OS

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

OS/400

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 path@ddcsmvs.lst blocking all
      sqlerror continue messages ddcsmvs.msg grant public
db2 connect reset
```

Où *DBALIAS*, *USERID* et *PASSWORD* s'appliquent au serveur de base de données hôte ou iSeries, *ddcsmvs.lst* est le fichier liste de liens de MVS et *path* indique l'emplacement du fichier liste de liens.

Par exemple, *drive:\sql11ib\bnd* s'applique à tous les systèmes d'exploitation Windows et *INSTHOME/sql11ib/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:\sql11ib\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
```

Remarques :

- 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 de ces fichiers de liste *ddcsxxx.lst* dans un serveur de base de données hôte ou iSeries devrait entraîner le signalement d'erreurs. Par exemple, lorsque vous effectuez des liaisons dans DB2 pour VM, plusieurs

messages d'avertissement peuvent être générés car DB2 pour VM n'autorise pas la déclaration des curseurs en tant que "WITH HOLD".

- b. Si vous vous connectez à une base de données DB2 via DB2 Connect, utilisez la liste de liens `db2ubind.lst` et ne spécifiez pas `sqlerror continue`, qui est uniquement valide lorsque vous vous connectez à un serveur de base de données hôte ou iSeries. 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.

Référence connexe :

- «Commande BIND» dans *Command Reference*
- «db2rbind - Commande Rebind all packages» dans *Command Reference*
- «Commande REBIND» dans *Command Reference*
- «Offres de produits DB2 Connect», à la page 3

Chapitre 7. Mises à jour multisites

Mises à jour multisites

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 UDB pour OS/390 et z/OS version 7
- DB2 pour z/OS version 8
- DB2 UDB pour iSeries requiert OS/400 version 5.1 ou version ultérieure

Une transaction répartie peut mettre à jour n'importe quelle combinaison de serveurs de base de données pris en charge. Par exemple, votre application peut mettre à jour diverses tables dans une base de données DB2 sous Windows, une base de données DB2 pour OS/390 et z/OS et une base de données DB2 UDB pour iSeries au cours d'une même transaction.

Concepts connexes :

- «Requêtes réparties», à la page 15
- «Mise à jour multisite et gestionnaire de points de synchronisation (SPM)», à la page 63
- «Unité d'oeuvre éloignée», à la page 13

Tâches connexes :

- «Activation des mises à jour multisites à l'aide du Centre de contrôle», à la page 62
- «Test d'une mise à jour multisite à l'aide du Centre de contrôle», à la page 63

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

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

Procédure :

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**.

Concepts connexes :

- «Mises à jour multisites», à la page 61

Tâches connexes :

- «Test d'une mise à jour multisite à l'aide du Centre de contrôle», à la page 63

Test d'une 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.

Procédure :

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.

Concepts connexes :

- «Mises à jour multisites», à la page 61

Tâches connexes :

- «Activation des mises à jour multisites à l'aide du Centre de contrôle», à la page 62

Mise à jour multisite et gestionnaire de points de synchronisation (SPM)

Les serveurs de base de données hôte ou iSeries requièrent DB2 Connect pour prendre part à une transaction répartie provenant d'applications Linux, Windows, UNIX et Web. En outre, plusieurs scénarios de mise à jour multisite impliquant des serveurs de base de données hôte ou iSeries requièrent la configuration du gestionnaire de points de synchronisation. 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 hôte ou iSeries à partir 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)	Hôte et base de données iSeries pris en charge
Oui	Oui	Serveur DB2 Connect DB2 Enterprise Server Edition avec licence DB2 Connect appliquée	DB2 UDB pour OS/390 et z/OS v7 DB2 UDB pour z/OS v8 ou version ultérieure
Non	Non	DB2 Connect Personal Edition Serveur DB2 Connect DB2 Enterprise Server Edition avec licence DB2 Connect appliquée	DB2 UDB pour OS/390 et z/OS v7 DB2 UDB pour z/OS v8 ou version ultérieure

Remarque : Une transaction répartie peut mettre à jour n'importe quelle combinaison de serveurs de base de données pris en charge. Par exemple, votre application peut mettre à jour plusieurs tables dans une base de données DB2 sous Windows, une base de données DB2 pour OS/390 et une base de données DB2 UDB pour iSeries, tout cela au cours d'une même transaction.

Concepts connexes :

- «Mises à jour multisites», à la page 61

Tâches connexes :

- «Configuration de DB2 Connect avec un gestionnaire de transactions compatible avec XA», à la page 64

Référence connexe :

- «Offres de produits DB2 Connect», à la page 3

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

Cette rubrique concerne les étapes de configuration requises pour l'utilisation des serveurs de base de données S/390, iSeries et zSeries dans votre moniteur TP.

Conditions préalables :

Vous possédez un moniteur TP opérationnel et vous avez installé DB2 Connect et configuré et testé une connexion au serveur de base de données hôte ou iSeries.

Procédure :

La configuration d'un accès vers un serveur de base de données DB2 LAN ne diffère pas de la configuration d'un accès vers un serveur de base de données hôte ou iSeries. Les instructions suivantes indiquent les étapes de configuration générales des moniteurs TP.

Pour configurer DB2 Connect en vue d'utiliser les serveurs de base de données S/390, iSeries et zSeries au sein de 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 ou iSeries ne prennent actuellement pas en charge l'interface XA.

Le SPM est un composant de DB2 Connect qui mappe le protocole de validation en deux phases XA au protocole de validation en deux phases utilisé par les serveurs de base de données hôte ou iSeries. 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 pour vous connecter à DB2 pour OS/390 et z/OS, ne modifiez aucun de ces paramètres par défaut. Dans ce cas de figure, aucune configuration SPM n'est requise car il s'agit déjà d'un paramètre optionnel.

Concepts connexes :

- «DB2 Connect et moniteurs de traitement de transactions», à la page 27
- «Support DB2 Connect pour les transactions à couplage lâche», à la page 65
- «Considérations sur la configuration des gestionnaires de transactions XA» dans *Administration Guide: Planning*

Support DB2 Connect pour les transactions à couplage lâche

Le support DB2 Connect pour les transactions à couplage lâche est destiné aux utilisateurs qui implémentent des applications réparties XA qui accèdent à DB2 UDB pour OS/390 et z/OS version 7 ou version ultérieure. Ce support permet à divers branchements de la même transaction globale de partager l'espace de verrouillage sous DB2 pour OS/390 et z/OS.

Le support pour les transactions à couplage lâche est conçu pour l'application 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. DB2 pour OS/390 et z/OS partage l'espace de verrouillage fourni et DB2 Connect envoie le XID à chaque connexion desservant divers branchements de la même transaction globale.

Concepts connexes :

- «Modèle de traitement réparti des transactions (DTP) X/Open» dans *Administration Guide: Planning*

Tâches connexes :

- «Mise à jour des serveurs de base de données hôte ou iSeries avec un gestionnaire de transactions compatible XA» dans *Administration Guide: Planning*

Chapitre 8. Mappage SQLCODE

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 de données hôte ou iSeries aux bons SQLCODE DB2.

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 hôte ou iSeries, tel que DB2 UDB pour OS/390 et z/OS, vous souhaitez peut-être désactiver le mappage SQLCODE. Ainsi, vous pouvez utiliser l'application sans modifier les SQLCODE qu'elle référence.

Tâches connexes :

- «Personnalisation du mappage SQLCODE», à la page 67
- «Désactivation du mappage de codes SQLCODE», à la page 67

Désactivation du mappage de codes SQLCODE

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 hôte iSeries, tel que DB2 UDB pour OS/390 et z/OS, vous souhaitez peut-être désactiver le mappage SQLCODE. Ainsi, vous pouvez utiliser l'application sans modifier les SQLCODE qu'elle référence.

Concepts connexes :

- «Mappage SQLCODE», à la page 67

Tâches connexes :

- «Personnalisation du mappage SQLCODE», à la page 67

Personnalisation du mappage SQLCODE

Par défaut, DB2 Connect mappe les SQLCODE et les jetons de chaque serveur de base de données hôte ou iSeries aux bons SQLCODE DB2. Les fichiers suivants sont des copies du mappage SQLCODE par défaut :

- dcs1dsn.map mappe les SQLCODE DB2 UDB pour OS/390 et z/OS.
- dcs1ari.map mappe les SQLCODE DB2 Server pour VSE & VM.
- dcs1qsq.map mappe les SQLCODE DB2 UDB pour iSeries.

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

Procédure :

Si vous souhaitez remplacer le mappage SQLCODE par défaut ou si vous utilisez un serveur de base de données hôte ou iSeries qui ne possède pas le 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.

Spécifiez le nom de fichier de votre nouveau fichier de mappage SQLCODE dans la chaîne de paramètres du répertoire DCS.

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_jetons]]
```

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

sqlcode Le SQLCODE du serveur de base de données hôte ou iSeries.

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 Le code de classe SQLSTATE du serveur de base de données hôte ou iSeries. *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*. Cette valeur est interprétée 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 être spécifiée avant le *n*, car le serveur de base de données hôte IBM ou iSeries prend uniquement en charge les jetons CHAR. Le caractère *i* doit être indiqué en minuscule.
- n* Un ou plusieurs numéros indiquant les jetons de serveur de base de données hôte ou iSeries 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 hôte ou iSeries, l'agencement indique l'ordre dans lequel les jetons seront placés dans la SQLCA.

Par exemple, le serveur de base de données hôte ou iSeries peut renvoyer deux jetons, 1 et 2. Si vous souhaitez que le jeton 2 apparaisse avant le jeton 1 dans la 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 7 indique un exemple de fichier de mappage SQLCODE.

```

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

-30021 , -30021 , (c1c,c2c)

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

```

Figure 7. Un fichier de mappage SQLCODE

Chaque instruction de mappage spécifiée dans le fichier est décrite ci-après :

1. Le SQLCODE est mappé de -007 vers -007. Le premier jeton d'entrée reçu du serveur de base de données hôte ou iSeries est utilisé en tant que premier jeton de sortie, et son type est défini par défaut sur 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 hôte ou iSeries est supprimé. 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 jetons reçus du serveur de base de données hôte ou iSeries 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 hôte ou iSeries 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 d'entrée reçus du serveur de base de données hôte ou iSeries sont de type CHAR et sont utilisés en tant que premier et second jetons dans la 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.

Concepts connexes :

- «Mappage SQLCODE», à la page 67

Tâches connexes :

- «Désactivation du mappage de codes SQLCODE», à la page 67

Chapitre 9. Moniteur du gestionnaire de base de données

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 système hôte ou iSeries, l'administrateur système peut déterminer si l'incident s'est produit sur le poste de travail DB2 Connect. Le moniteur du gestionnaire de base de données correspond :

- au 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**.

Concepts connexes :

- «Commutateurs du moniteur système» dans *System Monitor Guide and Reference*
- «Contrôle des performances à l'aide du moniteur de performances de Windows», à la page 74

Tâches connexes :

- «Paramétrage des inverseurs logiques du moniteur à partir d'une application client» dans *System Monitor Guide and Reference*
- «Paramétrage des inverseurs logiques du moniteur à partir de l'interpréteur de commandes (CLP)» dans *System Monitor Guide and Reference*

Référence connexe :

- «Offres de produits DB2 Connect», à la page 3

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.

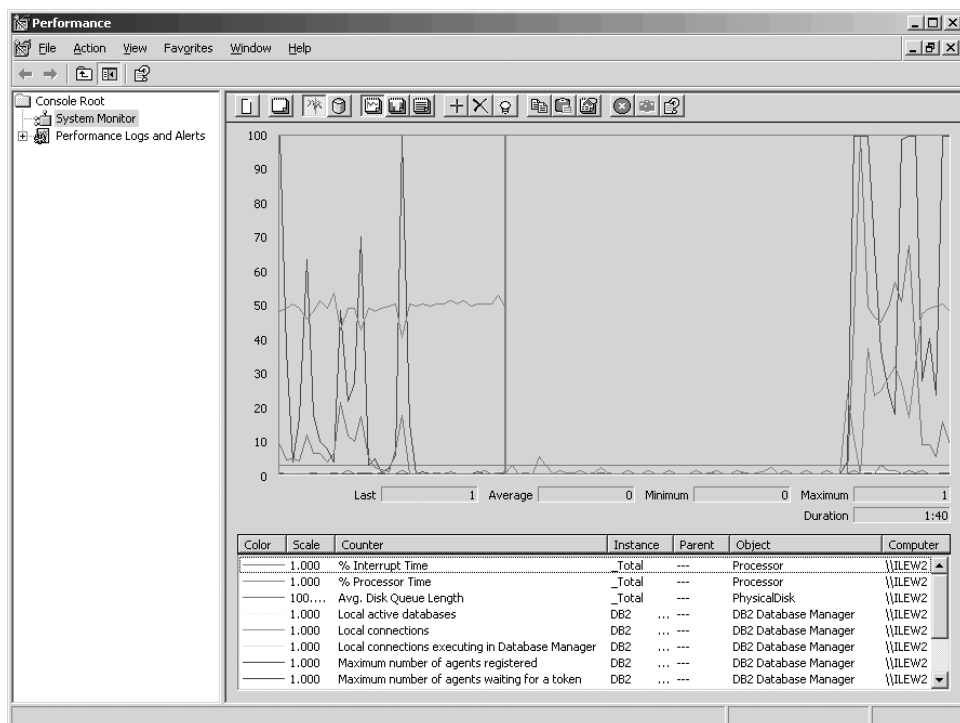


Figure 8. Moniteur de performances

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

Concepts connexes :

- «Contrôle des connexions des clients éloignés», à la page 73
- «Utilisation des commandes GET SNAPSHOT», à la page 75

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 images instantanées du moniteur DB2 peuvent être beaucoup plus détaillées. 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	= MVSDDB
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)	

Concepts connexes :

- «Contrôle des connexions des clients éloignés», à la page 73

Référence connexe :

- «Commande GET SNAPSHOT» dans *Command Reference*

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 les résultats qui suivent, le format de l'ID d'application hôte et de l'ID de l'application client peut varier en fonction de la version de l'hôte ou de la base de données iSeries et du niveau du support 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
Clients accédant aux serveurs de données avec un support de niveau du gestionnaire de base de données relationnelle 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 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 Universal Database pour z/OS et OS/390):

Auth Id	Application Name	Appl. Handle	Host Application Id
NEWTON	db2cli.exe	7	G91A0D3A.P8BC.060306212019
NEWTON	db2cli.exe	25	9.26.13.61.65289.060306213816
NEWTON	db2cli.exe	20	2002:91a:519:13:209:6bff:fe14:4fbb.7684.060306213741

Auth.Id

L'ID d'autorisation utilisé pour se connecter au serveur de base de données hôte ou iSeries. 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.

Host Application ID

L'une des situations suivantes s'est produite :

- 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 hôte ou iSeries. 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 entre les clients et le poste de travail DB2 Connect et entre le poste de travail DB2 Connect et le serveur de base de données hôte ou iSeries.

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 DB2 Universal Database version 7.1 et les produits et clients DB2 Connect.
- SQL08010 pour DB2 Universal Database version 8.1 et les produits et clients DB2 Connect.
- SQL08020 pour DB2 Universal Database version 8.2 et les produits et clients DB2 Connect.
- 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 la page de codes de la base de données (ou pour les bases de données du serveur de base de données hôte ou iSeries ou le CCSID du serveur de base de données hôte ou iSeries).

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 hôte ou iSeries (par exemple, DSN pour DB2 Universal Database pour z/OS et OS/390, ARI pour DB2 Server pour VSE & VM ou QSQ pour DB2 UDB pour iSeries)
- VV** Représente un numéro de version à deux chiffres, tel que 01.
- RR** Représente un numéro d'édition à deux chiffres.
- M** Représente un niveau de modification à un chiffre.

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. Cela signifie que la demande de connexion à une base de données hôte ou iSeries a été émise et que DB2 Connect attend que la connexion soit établie.
2. en attente de demande. Cette valeur indique que la connexion à la base de données hôte ou iSeries a été établie et que DB2 Connect attend une instruction SQL provenant de l'application client.
3. en attente de réponse. Cette valeur indique que l'instruction SQL a été envoyée à la base de données hôte ou iSeries.

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.

Référence connexe :

- «Commande LIST DCS APPLICATIONS» dans *Command Reference*
- «Commande LIST DCS DIRECTORY» dans *Command Reference*

Chapitre 10. Haute disponibilité

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 hôte et iSeries 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 ont besoin d'accéder aux données qui se trouvent sur plusieurs bases de données hôte et iSeries. 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.

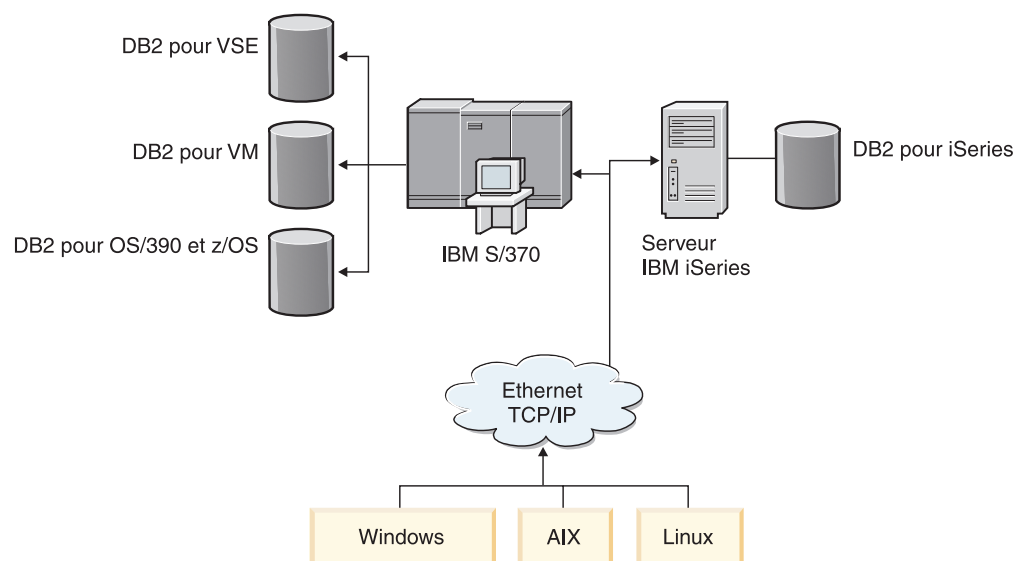


Figure 9. Exemple de scénario réseau

Pour que les postes de travail et les serveurs d'applications puissent accéder aux bases de données hôte et iSeries, vous avez besoin d'un composant de connectivité agissant en tant qu'intermédiaire. Ce composant doit offrir une connexion rapide, robuste et hautement disponible aux bases de données hôte et iSeries. Il doit également être évolutif afin d'anticiper toute croissance éventuelle du volume de connexions.

Les rubriques suivantes illustrent une solution utilisant DB2 Connect et la fonction de redirection automatique des clients :

- Description et configuration de la redirection automatique des clients
- Remarques relatives au composant Distributor

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89
- «Conversion de données sur l'hôte», à la page 118
- «Redirection automatique du client - Description et configuration», à la page 84

Référence connexe :

- «Remarques du distributeur», à la page 86

Redirection automatique du client - Description et configuration

Le principal objectif de la fonction de redirection automatique du client est de permettre à une application client de base de données DB2 d'effectuer une reprise après une perte de communication afin que l'application poursuive son travail avec un minimum d'interruption. Comme son nom l'indique, la redirection est essentielle à la continuité des opérations. Mais elle est possible uniquement lorsqu'un emplacement de secours a été identifié pour la connexion client.

La fonction de redirection automatique du client s'utilise dans les environnements pouvant être configurés, suivants :

1. Enterprise Server Edition (ESE) avec la fonction de partitionnement de base de données (DPF)
2. DataPropagator (DPROPR)- Réplication de style
3. High Availability Cluster Multiprocessor (HACMP)
4. High Availability Disaster Recovery (HADR).

La redirection automatique du client fonctionne en parallèle avec la fonction HADR afin de permettre à une application client de poursuivre son activité avec un minimum d'interruption après la reprise de la base de données à laquelle elle avait accès.

Dans le cas du serveur DB2 Connect, puisqu'aucune exigence de synchronisation des bases de données locales n'existe, vous devez uniquement vous assurer que la base de données hôte ou iSeries cible des deux serveurs DB2 Connect, l'original et le serveur de secours, est cataloguée de façon à être accessible à l'aide d'un alias de base de données identique.

Pour que le système de base de données DB2 puisse effectuer une reprise après une perte de communication, vous devez indiquer l'emplacement d'un serveur de secours avant que la perte ne survienne. La commande **UPDATE ALTERNATE SERVER FOR DATABASE** permet de définir l'emplacement du serveur de secours sur une base de données particulière. Le nom d'hôte et le numéro de port de secours sont donnés dans le cadre de la commande. L'emplacement est stocké dans le fichier du répertoire système des bases de données du serveur. Afin de vous assurer que l'emplacement du serveur de secours indiqué s'applique à tous les clients, il doit être spécifié côté serveur. Le serveur de secours est ignoré s'il est défini sur l'instance de client.

Par exemple, supposons qu'une base de données se trouve sur la partition de base de données appelée «N1» (avec un nom d'hôte XXX et un numéro de port YYY).

L'administrateur de base de données souhaite définir l'emplacement du serveur de secours sur le nom d'hôte = AAA avec un numéro de port 123. Voici la commande que devra exécuter l'administrateur de base de données sur la partition de base N1 (sur l'instance de serveur) :

```
db2 update alternate server for database db2 using hostname AAA port 123
```

Après avoir indiqué l'emplacement du serveur de secours sur une base de données spécifique au niveau de l'instance de serveur, les informations sur l'emplacement sont renvoyées au client dans le cadre du processus de connexion. Si la communication entre le client et le serveur est perdue pour une raison ou une autre, le client DB2 codé va tenter de rétablir la connexion à partir des informations relatives au serveur de secours. Le client DB2 va tenter de se reconnecter au serveur d'origine et au serveur de secours, alternant les tentatives entre les deux. Ces tentatives commencent avec un rythme très rapide et les intervalles entre les tentatives se rallongent graduellement.

Dès qu'une connexion aboutit, le code SQLCODE -30108 est renvoyé pour indiquer qu'une connexion a été rétablie avec la base de données suite à l'incident de communication. L'adresse IP du nom d'hôte et le numéro de port du nom de service sont renvoyés. Le code client renvoie uniquement l'erreur de l'incident de communication d'origine vers l'application si le rétablissement des communications client est impossible sur l'un ou l'autre des serveurs d'origine ou de secours.

Vous devez prendre en considération les deux éléments suivants qui impliquent la connectivité au serveur de secours avec le serveur DB2 Connect :

- La première remarque concerne l'utilisation du serveur DB2 Connect pour la fourniture d'accès vers une base de données hôte ou iSeries pour le compte des clients distant et local. Dans de telles situations, les informations de connectivité du serveur de secours peuvent être confuses dans une entrée de répertoire système des bases de données. Pour mettre fin à cette confusion, vous devez cataloguer deux entrées dans un répertoire système de bases de données pour représenter la même base de données hôte ou iSeries. Cataloguez une entrée pour le client distant et une autre pour le client local.
- Deuxièmement, les informations relatives au serveur de secours qui sont renvoyées par un serveur cible sont conservées uniquement dans la mémoire cache. Si le processus DB2 s'arrête, les informations de la mémoire cache, par conséquent les informations du serveur de secours, sont perdues.

En règle générale, si le serveur de secours est spécifié, la redirection automatique du client sera activée quand une erreur de communication (sqlcode -30081) ou un code sqlcode -1224 est détecté. Toutefois, dans un environnement HADR (High Availability Disaster Recovery), elle sera également activée si un code sqlcode -1776 est renvoyé d'un serveur de secours HADR.

Concepts connexes :

- «Limitations de la redirection automatique des clients» dans *Administration Guide: Implementation*
- «Configuration de la redirection du client lors de l'utilisation de pilotes JCC Type 4» dans *Administration Guide: Implementation*

Référence connexe :

- «Exemples de redirection automatique des clients» dans *Administration Guide: Implementation*

- «Présentation de la redirection client automatique» dans *Administration Guide: Implementation*

Remarques du distributeur

Quand une connexion client-serveur échoue, les demandes de nouvelle connexion du client sont réparties dans un ensemble défini de systèmes par un distributeur ou un répartiteur, tel que WebSphere EdgeServer.

Vous pouvez utiliser une technologie de distributeur dans des environnements semblables à ceux qui suivent :

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

où :

- Le composant de la technologie de distributeur dispose d'un nom d'hôte TCP/IP **DThostname**
- Le serveur 1 DB2 Connect dispose d'un nom d'hôte TCP/IP **GWYhostname1**
- Le serveur 2 DB2 Connect dispose d'un nom d'hôte TCP/IP **GWYhostname2**
- Le serveur DB2 z/OS dispose d'un nom d'hôte TCP/IP **zOShostname**

Le client est catalogué avec le nom **DThostname** afin d'utiliser la technologie de distributeur pour accéder à l'un des serveurs DB2 Connect. La technologie de distributeur qui intervient décide d'utiliser **GWYhostname1** ou **GWYhostname2**. Une fois la décision prise, le client dispose d'une connexion socket directe à l'une de ces deux passerelles DB2 Connect. Une fois la connectivité socket établie vers le serveur DB2 Connect choisi, vous disposez d'un client standard pour le serveur DB2 Connect pour la connectivité DB2 z/OS.

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

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

Le distributeur n'effectue aucune nouvelle tentative de connexion en cas d'incident de communication. Pour activer la fonction de redirection automatique du client d'une base de données dans un tel environnement, le serveur de secours des ou de la base de données associée au serveur DB2 Connect (serveur 1 DB2 Connect ou serveur 2 DB2 Connect) doit être défini comme étant le distributeur (**DThostname**). Ensuite, si le serveur 1 DB2 Connect se verrouille pour une raison ou une autre, la redirection automatique du client est déclenchée et une nouvelle tentative de connexion est lancée avec le distributeur à la fois comme serveur principal et serveur de secours. Cette option permet de regrouper et de gérer les fonctionnalités du distributeur avec la fonction de redirection automatique du client DB2. Si le serveur de secours est défini comme serveur hôte avec un nom d'hôte différent de celui du distributeur, il offre néanmoins la fonction de redirection automatique du client. Toutefois, les clients vont établir des connexions directes au serveur de secours et ignorer la technologie du distributeur, qui élimine le distributeur et la valeur qu'il apporte.

La fonction de redirection automatique du client intercepte les codes SQL suivants :

- sqlcode -20157

- sqlcode -1768 (code raison = 7)

Remarque : La fonction de redirection du client risque de ne pas être informée en temps utile des échecs de socket si le paramètre de configuration du système d'exploitation "Signal de présence TCP" est trop élevé. (Le nom de ce paramètre de configuration varie en fonction de la plateforme.)

Référence connexe :

- «Présentation de la redirection client automatique» dans *Administration Guide: Implementation*

Chapitre 11. Performances

Considérations sur les performances de DB2 Connect

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 hôte ou iSeries 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 10, à la page 90 illustre le chemin d'accès au flot de données entre le serveur de base de données hôte ou iSeries et le poste de travail via DB2 Connect.

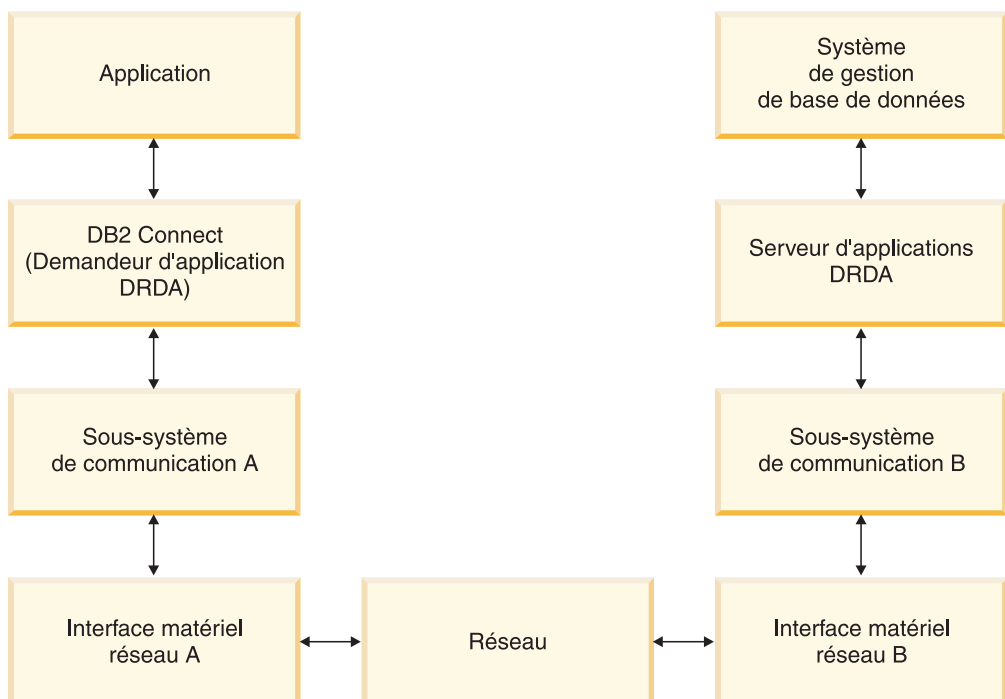


Figure 10. Flots de données dans DB2 Connect

- La base de données hôte ou iSeries ainsi qu'une partie du sous-système de communication B fonctionnent 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 B peut se composer de contrôleurs de communication, tels que 3745 ou 3172, ou d'une carte de réseau en anneau à jeton pour un serveur iSeries. 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, 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 12. 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 et glance, si disponible	
Windows	Moniteur de performances Microsoft	

Tableau 13. 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.
OS/390 ou zSeries	DB2PM (IBM), OMEGAMON/DB2 (Candle), TMON (Landmark), INSIGHT (Goal Systems) et DB2AM (BMC)	
Windows	Moniteur de performances Microsoft	

Tableau 14. 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.

Concepts connexes :

- «Conception d'application», à la page 92
- «Regroupement de connexions», à la page 95
- «Optimisation de DB2 Connect», à la page 107

Tâches connexes :

- «Optimisation de l'accès ODBC», à la page 113

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 le temps système de l'unité centrale du serveur de base de données hôte et iSeries et 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 d'interprétation, tel que le langage REXX, sont significativement plus lentes que les mêmes instructions SQL dans un langage de compilation, tel que le langage 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.

Concepts connexes :

- «Regroupement de connexions», à la page 95
- «Considérations sur les performances de DB2 Connect», à la page 89

Gestion des connexions

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. Tous agents mis en veille est conservé dans un regroupement d'agents en veille. Vous pouvez déterminer 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.

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 maximal d'agents pouvant être actifs de manière concurrente 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.)

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.

Concepts connexes :

- «Concentrateur de connexion», à la page 98
- «Regroupements de connexions et concentrateur de connexion», à la page 103
- «Considérations sur les performances de DB2 Connect», à la page 89

Concentrateur de connexion

Le concentrateur de connexion réduit le nombre de ressources requises sur les serveurs de base de données DB2 pour OS/390 et z/OS afin de prendre en charge de grands nombres de postes de travail et d'utilisateurs Web. Cette fonction peut augmenter de manière significative l'évolutivité de votre solution DB2 pour OS/390 et z/OS et DB2 Connect tout en offrant la répartition des charges pour garantir des opérations et des niveaux de transaction insensibles aux pannes dans des environnements de partage de données DB2 pour OS/390 et z/OS.

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 du *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 significativement les ressources nécessaires sur l'hôte S/390 ou les serveurs de base de données iSeries. Tout cela est rendu possible grâce à la concentration des charges de travail provenant de toutes les applications dans un plus petit nombre de connexions au serveur de base de données hôte S/390 ou iSeries. 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 met en oeuvre un algorithme d'ordonnement sophistiqué qui utilise les informations du Work Load Manager (WLM) OS/390 et z/OS. 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 réallouer de manière transparente la charge de travail des membres défectueux ou surchargés à des membres fonctionnels et sous-utilisés. Le concentrateur de connexion DB2 Connect est activé lorsque vous définissez le nombre maximal d'agents logiques (*max_connections*) sur une valeur supérieure 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 recherchait 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. C'est-à-dire que 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 lorsque le nombre d'agents logiques dépasse la limite de descripteurs de fichier.

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 capable d'utiliser la totalité de l'ensemble des fonctions du concentrateur.
- Si vous déclarez des tables temporaires générales, elles doivent être fermées de manière explicite au terme de la transaction ou du branchement. Le défaut de fermeture des tables entraînera la désactivation du concentrateur de connexion sans perturber le fonctionnement de l'application.
- 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.

Dans DB2 version 9, ou version 8 FixPak 13, (ou supérieure), la prise en charge du concentrateur DB2 Connect ne peut être activée qu'avec iSeries version 5.4 (PTF SI23726). Sinon, seule la partie XQ du concentrateur de connexion est prise en charge.

Activation du concentrateur de connexion :

Le paramètre de configuration du gestionnaire de base de données MAX_CONNECTIONS 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 dispose 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. MAX_CONNECTIONS sera défini sur un intervalle numérique compris entre MAX_COORDAGENTS et 64.000. Le nombre par défaut d'agents logiques est égal à MAX_COORDAGENTS.

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

MAXAGENTS

Nombre maximal d'agents exécutants.

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 paramètre MAXAGENTS ou sur le nombre moyen de clients.

NUM_INITAGENTS

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

Prise en charge des transactions XA :

L'architecture du concentrateur de connexion permet à DB2 Connect de prendre en charge les transactions XA à configuration groupée vers DB2 pour OS/390 et z/OS et DB2 pour iSeries. 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. Considérez un environnement dans lequel 4.000 connexions concurrentes, voire plus, sont nécessaires. 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.

Le système de serveur DB2 Connect ne pourra peut-être pas gérer les 4000 connexions ouvertes simultanées à la machine de la 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_AGENTS       = 1000
MAX_COORDAGENTS = 1000
NUM_POOLAGENTS  = 1000
```

Le concentrateur conservera les 4000 connexions concurrentes ouvertes, même si la passerelle gère uniquement 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 que le moniteur TP est utilisé avec une passerelle DB2 Connect et une base de données zSeries ou iSeries. 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.

Concepts connexes :

- «Regroupement de connexions», à la page 95
- «Regroupements de connexions et concentrateur de connexion», à la page 103

- «Considérations sur les performances de DB2 Connect», à la page 89

Regroupements de connexions et concentrateur de connexion

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 aide à améliorer l'évolutivité de votre solution DB2 pour OS/390 et 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.

Concepts connexes :

- «Concentrateur de connexion», à la page 98
- «Regroupement de connexions», à la page 95
- «Considérations sur les performances de DB2 Connect», à la page 89

Support Sysplex DB2 Connect

Support Sysplex DB2 Connect

Un sysplex est un ensemble de serveurs zSeries qui coopèrent pour le traitement tant au niveau matériel que logiciel. 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.

Sysplex permet à DB2 Connect de transférer de façon transparente une connexion entrante provenant d'un serveur de base de données éloigné vers un serveur de sauvegarde déterminé, au cas où le premier serveur aurait une défaillance. La prise en charge de Sysplex par DB2 Connect est activée par défaut, toutefois chaque entrée du catalogue de base de données DCS doit être configurée pour permettre le support Sysplex.

Grâce à la redirection automatique des clients, le comportement par défaut d'une connexion activée d'un sysplex est de procéder à une nouvelle tentative en cas d'incident de communication. Cependant, les instructions **SET** ne sont pas renvoyées lorsque la redirection des clients est activée dans DB2 pour z/OS. Pour contourner cette limitation, les applications doivent réinitialiser leur environnement d'exécution elles-mêmes.

Vous pouvez configurer le comportement de nouvelle tentative, y compris la désactivation, à l'aide des variables de registre `DB2_MAX_CLIENT_CONNRETRIES` et `DB2_CONNRETRIES_INTERVAL`.

Concepts connexes :

- «Configuration requise pour Sysplex», à la page 105
- «Considérations sur l'exploitation de OS/390 et zSeries SYSPLEX», à la page 104
- «Exploitation de Sysplex avec DB2», à la page 105
- «Redirection automatique du client et reprise à haut niveau de disponibilité (HADR)» dans *Data Recovery and High Availability Guide and Reference*

Référence connexe :

- «Configuration de la redirection client automatique (`DB2_MAX_CLIENT_CONNRETRIES` et `DB2_CONNRETRIES_INTERVAL`)» dans *Administration Guide: Implementation*

Considérations sur l'exploitation de OS/390 et zSeries SYSPLEX

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 pour OS/390 et z/OS et que ce dernier fonctionne dans un environnement de partage des données, DB2 Connect répartit la charge entre les différents sous-systèmes DB2 que comprend le groupe de partage des données, en fonction des informations relatives à la charge du système 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 DDF (Distributed Data Facility) OS/390 et z/OS 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.

La liste de serveurs devient inaccessible s'il n'y a pas d'agent et aucune connexion à une base de données. Par conséquent, vous devez conserver au minimum un agent pour gérer la liste de serveurs Sysplex. Activez le regroupement de connexions en exécutant les commande suivantes :

```
db2 update dbm cfg using num_poolagents number
db2stop
db2start
```

où *number* représente le nombre maximal d'agents autorisés dans le regroupement dans l'instance DB2. Le regroupement de connexions est activé lorsque la valeur de *number* est supérieure à 0. Vous pouvez également définir le num_poolagents sur -1, ce qui donne la moitié de la valeur affectée au paramètre de configuration maxagents.

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.

Concepts connexes :

- «Configuration requise pour Sysplex», à la page 105
- «Support Sysplex DB2 Connect», à la page 103
- «Exploitation de Sysplex avec DB2», à la page 105

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.

Concepts connexes :

- «Considérations sur l'exploitation de OS/390 et zSeries SYSPLEX», à la page 104
- «Support Sysplex DB2 Connect», à la page 103
- «Exploitation de Sysplex avec DB2», à la page 105

Exploitation de Sysplex avec DB2

Pour prendre un exemple classique, le serveur A DB2 Connect converse avec un Sysplex qui contient deux serveurs DB2 pour OS/390 et 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 de Sysplex étant active à la fois pour le serveur DB2 Connect et pour l'entrée d'annuaire DCS, DB2 pour OS/390 et z/OS identifie auprès de DB2 Connect l'adresse réseau de 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 des adresses fournie par DB2 pour OS/390 et z/OS comprend également des informations de priorité, y compris 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 placée dans la mémoire 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 fonction des informations de priorité renvoyées par DB2 pour OS/390 et 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 la tentative de connexion échoue, d'autres adresses sont essayées successivement telles qu'elles ont été renvoyées par DB2 pour OS/390 et z/OS, par ordre de priorité décroissante. 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.

Concepts connexes :

- «Configuration requise pour Sysplex», à la page 105
- «Considérations sur l'exploitation de OS/390 et zSeries SYSPLEX», à la page 104
- «Support Sysplex DB2 Connect», à la page 103

Référence connexe :

- «db2pd - Commande Monitor and troubleshoot DB2 database» dans *Command Reference*

Optimisation de DB2 Connect

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 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 :

AGENTPRI s'applique uniquement aux clients distants. AGENTPRI contrôle la priorité donnée par le planificateur du système d'exploitation aux agents d'une instance DB2 Connect. L'instance DB2 Connect reçoit davantage de cycles de processeur et possède une priorité plus élevée (numéro inférieur). Le nombre de cycles du processeur consacré aux autres processus s'exécutant sur le poste de travail DB2 Connect diminue. Par exemple, vous pouvez posséder une instance DB2 Connect à priorité élevée et une instance DB2 Connect à priorité faible sur le même poste de travail avec des valeurs AGENTPRI différentes.

Chaque connexion établie d'une machine client vers un serveur de base de données hôte ou iSeries via DB2 Connect requiert un agent fonctionnant sur le poste de travail DB2 Connect. Définissez MAXAGENTS sur une valeur supérieure ou égale au nombre maximal moyen de connexions client éloignées accédant à un serveur de base de données hôte ou iSeries via le poste de travail DB2 Connect.

Pour améliorer les performances, NUM_POOLAGENTS doit être configuré sur une valeur égale au paramètre MAXAGENTS ou sur le nombre moyen de clients.

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 DB2 Driver pour JDBC et SQLJ

com.ibm.db2.jcc.DB2BaseDataSource.clientAccountingInformation
(propriété)

DB2 .NET Data Provider

DB2Connection.ClientAccountingInformation (propriété)

CLI/ODBC

ClientAcctStr (mot clé de configuration 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.

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89
- «Optimisation de la base de données hôte», à la page 109

Optimisation de la base de données hôte

Les performances du système seront affectées par les performances du serveur de base de données hôte ou iSeries. 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. Consultez la documentation relative aux performances système du serveur de base de données hôte ou iSeries pour obtenir des informations supplémentaires.

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).

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89
- «Considérations relatives à l'optimisation du réseau», à la page 109

Considérations relatives à l'optimisation du 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 15.

Tableau 15. Méthodes de réglage des performances réseau

Éléments 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.

Tableau 15. Méthodes de réglage des performances réseau (suite)

Éléments à rechercher	Exemple	Paramètre	Remarques
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.
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.

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89
- «Conflit des ressources système», à la page 111

Conflit des 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 appairer 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 ?

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89
- «Résolution des incidents de performances de DB2 Connect», à la page 112

Résolution des incidents de performances de DB2 Connect

Si les utilisateurs DB2 Connect connaissent de longs temps de réponse pour les grandes requêtes issues des serveurs hôte ou iSeries, les zones suivantes doivent être examinées à la recherche de l'origine probable des problèmes de performance :

1. Pour les requêtes qui renvoient de grands blocs de données du serveur hôte ou iSeries (généralement 32 de données et plus), vérifiez que le paramètre de configuration du gestionnaire de base de données RQRIOBLK est défini sur 32767. Effectuez cette opération à l'aide de l'interpréteur de commandes (CLP) en procédant comme suit :

```
db2 update database manager configuration using RQRIOBLK 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 à l'aide de matériel Ethernet, prenez note de la taille de trame maximale Ethernet de 1536 octets qui peut être un facteur limitatif.

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89

Optimisation de DB2 pour OS/390 et z/OS

Vous pouvez optimiser le traitement par unité d'exécution dans OS/390 et 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.

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89

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.

Procédure :

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.

Concepts connexes :

- «Conception d'application», à la page 92
- «Optimisation des performances d'applications CLI/ODBC», à la page 114
- «Considérations sur les performances de DB2 Connect», à la page 89

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

Tâches connexes :

- «Optimisation de l'accès ODBC», à la page 113
- «Appel de procédures mémorisées à partir d'applications CLI» dans *Call Level Interface Guide and Reference, Volume 1*

Référence connexe :

- «Mot clé de configuration SysSchema CLI/ODBC» dans *Call Level Interface Guide and Reference, Volume 1*

Augmentation des taux de transfert de données DB2 Connect

Outre la capacité de bloquer des lignes d'un ensemble de résultats de requêtes, DB2 pour OS/390 et 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, tel que DB2 Connect. Au lieu d'envoyer des requêtes vers le serveur DB2 pour OS/390 et z/OS qui demande un bloc de données de ligne à la fois, le client peut désormais, s'il le souhaite, demander que le serveur renvoie certains blocs de requêtes 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 partir d'un serveur DB2 pour OS/390 et 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.

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89
- «Bloc de requête supplémentaire», à la page 116
- «Mise à l'échelle des fenêtres RFC-1323», à la page 117

Bloc de requête supplémentaire

La prise en charge de blocs de requête supplémentaires sur les serveurs sur lesquels DB2 UDB pour OS/390 et z/OS version ou version ultérieure est installé, 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.

Du côté client, côté sur lequel l'application accède à DB2 pour z/OS directement via une installation DB2 Connect colocalisée ou via une installation serveur DB2 Connect séparée, il existe divers procédés 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 pour OS/390 et z/OS tente de bloquer le nombre souhaité de lignes à renvoyer à DB2 Connect, sujet du paramètre d'installation EXTRA BLOCKS SRV DDF. L'application peut choisir d'extraire plus de N lignes car DB2 pour z/OS ne limite pas le nombre de lignes pouvant être renvoyées dans l'ensemble de résultats de la requête défini sur N.
- La clause 'FETCH FIRST N ROWS ONLY' fonctionne de manière similaire, à l'exception que l'ensemble de résultats de la requête est limité à N lignes par DB2 pour OS/390 et 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 plutôt utilisée pour un serveur DB2 UDB pour OS/390 et 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 pour OS/390 et 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 OS/390 et z/OS 6.x. DB2 Connect balise également la clause 'FETCH FIRST N ROWS ONLY' pour un serveur DB2 pour z/OS 7.1 ou version supérieure.

Concepts connexes :

- «Augmentation des taux de transfert de données DB2 Connect», à la page 115
- «Considérations sur les performances de DB2 Connect», à la page 89
- «Mise à l'échelle des fenêtres RFC-1323», à la page 117

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 option 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 pour z/OS, la mise à l'échelle des fenêtres peut actuellement être uniquement activée via le système d'exploitation en définissant `TCPRCVBUFRSIZE` sur n'importe quelle valeur supérieure à 64 K. Si vous utilisez un client DB2 pour accéder à une base de données hôte ou iSeries DB2 via un poste de travail serveur DB2 Connect, vous pouvez activer la mise à l'échelle des fenêtres sur le client également. Avec le même jeton, vous pouvez également activer la mise à l'échelle des fenêtres entre un client DB2 éloigné et un serveur DB2 de poste de travail lorsqu'aucune base de données hôte ou iSeries 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. Pour consulter une introduction sur le réglage du réseau afin d'améliorer les performances, consultez le site <http://www.networking.ibm.com/>.

Concepts connexes :

- «Augmentation des taux de transfert de données DB2 Connect», à la page 115
- «Considérations sur les performances de DB2 Connect», à la page 89
- «Bloc de requête supplémentaire», à la page 116

Conversion de données sur l'hôte

Lors de transferts d'informations entre divers environnements (tels que Intel [Windows], IEEE [systèmes d'exploitation Linux et UNIX], zSeries [VM, VSE, z/OS], iSeries [OS/400]), des types de données numériques (tels que les décimales, les entiers ou les virgules flottantes) 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 hôte ou iSeries.

Concepts connexes :

- «Types de données pour les données de type caractères», à la page 118
- «Considérations sur les performances de DB2 Connect», à la page 89

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.

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89

- «Conversion de données sur l'hôte», à la page 118

Matériel réseaux

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

La vitesse de transfert de données est limitée par le support de transmission le plus lent dans le chemin d'accès au serveur de base de données hôte ou iSeries.

- 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.

Concepts connexes :

- «Considérations sur les performances de DB2 Connect», à la page 89

Chapitre 12. Identification des incidents

Identification d'incident

L'environnement DB2 Connect implique divers logiciels et matériels et diverses communications. La meilleure approche d'identification des incidents consiste en un procédé d'élimination et d'amélioration des données disponibles afin d'arriver à une conclusion (l'emplacement de l'erreur).

Après avoir rassemblé les informations pertinentes et selon votre sélection de la rubrique applicable, passez à la section référencée.

Concepts connexes :

- «Outils de diagnostic», à la page 122
- «Regroupement d'informations adéquates», à la page 121
- «Connexion initiale non aboutie», à la page 122
- «Incidents rencontrés après une connexion initiale», à la page 123
- «Utilitaire de trace», à la page 124

Concepts de détermination d'incident

Regroupement d'informations adéquates

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 ?
- La panne de communication a-t-elle lieu entre le client et le poste de travail DB2 Connect, le poste de travail DB2 Connect et le serveur de base de données hôte ou iSeries, avec tous les clients ou avec un seul client ?
- Que vous pouvez déduire du contenu du message et des jetons renvoyés dans le message ?
- L'utilisation des outils de diagnostic 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 ?

Concepts connexes :

- «Outils de diagnostic», à la page 122

- «Identification d'incident», à la page 121

Outils de diagnostic

Lorsque vous rencontrez un problème, vous pouvez procéder comme suit :

- Le premier journal de maintenance des incidents dans lequel les données de diagnostic sont rassemblées et stockées dans un format lisible est conservé dans le journal de notification de l'administration.
- Les deux journaux se trouvent dans le chemin d'accès spécifié :
Ce fichier se trouve dans le chemin `/u/db2/sqllib/db2dump/notifyloglevel.nfy` sur les systèmes UNIX, où `db2` indique le nom de l'instance.
Ce fichier se trouve dans le chemin `x:\sqllib\db2\db2diag.log` sur les systèmes Windows, où `x:` indique l'unité logique et `db2` le nom de l'instance.
- Pour les systèmes d'exploitation Windows, vous pouvez utiliser l'Observateur d'événements pour visualiser le journal de notification de l'administration.
- L'utilitaire de trace
- 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.

Concepts connexes :

- «Résolution des incidents de performances de DB2 Connect», à la page 112
- «Utilitaire de trace», à la page 124

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 hôte ou iSeries et du poste de travail ont-elles été configurées correctement ?*
 - Trois configurations doivent être prises en considération :
 - a. La configuration du serveur de base de données hôte ou iSeries identifie le demandeur d'application sur le serveur. Le système de gestion du

serveur de base de données hôte ou iSeries possédera des entrées de catalogue système qui définiront 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 du serveur et du serveur hôte ou iSeries vers le 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 hôte ou iSeries et les administrateurs de réseau possèdent des fonctionnalités de diagnostic d'incidents.
5. *Possédez-vous les droits d'accès requis par le système de gestion de base de données du serveur hôte ou iSeries pour utiliser la base de données du serveur hôte ou iSeries ?*
 - 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 hôte ou iSeries, y parvenez-vous ou pas ?*
 - Avez-vous suivi la procédure de liaison de l'interpréteur de commandes à un serveur de base de données hôte ou iSeries ?

Concepts connexes :

- «Identification d'incident», à la page 121
- «Incidents rencontrés après une connexion initiale», à la page 123

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 ?
 - Une modification provisoire du logiciel a-t-elle été récemment installée ? Si l'incident se produit lorsqu'un utilisateur tente d'utiliser une fonction qui n'a plus été utilisée (ou chargée) sur leur système d'exploitation depuis son installation, déterminez le niveau de modification provisoire du logiciel IBM le plus récent et chargez ce niveau *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 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. *La commande DB2START a-t-elle été exécutée sur le 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 ne parvenait 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.

Remarque : DB2 Connect ne prend pas en charge la commande PING lorsqu'elle est exécutée à partir d'un client version 7 par l'intermédiaire d'un serveur DB2 Connect version 9.

Concepts connexes :

- «Identification d'incident», à la page 121
- «Utilitaire de trace», à la page 124

Utilitaire de trace

L'utilitaire **db2drdat** enregistre les données interchangeées entre le serveur DB2 Connect (à la demande du client de base de données) et le serveur de base de données hôte ou iSeries.

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. Par exemple, si vous exécutez une instruction de base de donnéesCONNECT TO pour un serveur de base de données hôte ou iSeries et que la commande échoue et que vous recevez

un code retour infructueux. Si vous comprenez exactement les informations qui ont été acheminées vers le système de gestion du serveur de base de données hôte ou iSeries, vous serez peut-être capable de déterminer l'origine de l'échec même si les informations relatives au code retour sont générales. De nombreux incidents sont provoqués par de simples erreurs de l'utilisateur.

Le résultat issu 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 hôte ou iSeries. Les données envoyées au serveur de base de données hôte ou iSeries s'intitulent SEND BUFFER et les données reçues du serveur de base de données hôte ou iSeries 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. Le champ SQLCODE d'une SQLCA est la valeur *unmapped* renvoyée par le serveur de base de données hôte ou iSeries. 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.

Concepts connexes :

- «Sortie de trace», à la page 125
- «Analyse du fichier de sortie de trace», à la page 126

Référence connexe :

- «db2drdat - Commande DRDA Trace» dans *Command Reference*

Caractéristiques de l'utilitaire de trace

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.

Remarques :

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.

Concepts connexes :

- «Analyse du fichier de sortie de trace», à la page 126
- «Utilitaire de trace», à la page 124

Référence connexe :

- «db2drdat - Commande DRDA Trace» dans *Command Reference*

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 hôte ou iSeries
- Le système de gestion du serveur de base de données hôte ou iSeries avec lequel le système DB2 Connect communique.

La première mémoire tampon contient les commandes Exchange Server Attributes (EXCSAT) et Access RDB (ACCRDB) envoyées au système de gestion du serveur de base de données hôte ou iSeries. 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 hôte ou iSeries. 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. Au sein de 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 (Nom externe), souvent placé dans les données de diagnostic du système de gestion de la base de données hôte ou iSeries. 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 de base de données Linux ou UNIX, cette valeur peut être mise en corrélation avec la commande **ps** qui renvoie des données relatives à l'état des processus, qu'il s'agisse de processus actifs ou de sorties 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' qui représente 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 hôte ou iSeries dans le message de réponse Access RDB (ACCRDBRM) de la seconde mémoire tampon. Cette mémoire tampon contient les données EXCSATRD suivies du message ACCRDBRM. L'exemple de fichier de sortie contient les deux valeurs de CCSID du système du serveur de base de données hôte ou iSeries. 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 renvoyée par le serveur de base de données hôte ou iSeries, le SQLCODE -332 sera renvoyé à l'utilisateur avec les pages de codes source et cible. Si le serveur de base de données hôte ou iSeries ne reconnaît pas le jeu de codes envoyé 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 traduit en code 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. Selon les standards, le DSN est DB2 Universal Database pour z/OS et OS/390. Le numéro de version est également indiqué. L'ARI est DB2 Server pour VSE & VM, le SQL est la base de données DB2 ou DB2 Connect et le QSQ est DB2 UDB pour iSeries.

Concepts connexes :

- «Sortie de trace», à la page 125
- «Utilitaire de trace», à la page 124

Référence connexe :

- «db2drdat - Commande DRDA Trace» dans *Command Reference*
- «Informations de mémoire tampon postérieures pour les traces DRDA», à la page 133
- «Modèles de fichiers de sortie de trace», à la page 128

Modèles de fichiers de sortie de trace

Les figures suivantes affichent un exemple de résultats illustrant certains flots de données DRDA entre des postes de travail DB2 Connect et un serveur de base de données hôte ou iSeries. 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 11 utilise DB2 Connect Enterprise Edition version 9.1 et DB2 UDB pour 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
```

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 1 de 20)

```
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	00	C3D0	41	00	10	00	BD		10	41	00	7F	11	5E	84	82	...	A.....A...^..	.C}.....";db
0010	F2	29	74	04	04	04	04	04	04	04	04	04	04	04	04	04	...	@@@@@@@@@@@@@@	2bp
0020	40	40	F0	F0	F0	C3	C3	F0	F0	F0	00	00	00	00	00	00	@@.....	000C50CC000...	
0030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0040	00	00	00	00	00	00	00	00	00	00	60	F0	F0			-00	
0050	F0	F1	A2	A4	95	40	40	40	40	40	40	40	40	40	40	40	@@@@@@@@@@@@	01sun
0060	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	@@@@@@@@@@@@@@@@
0070	C4	C5	C3	E5	F8	40	40	F0	A2	A4	95	40	40	40	40	40	@@...@@@	DECV8 0sun
0080	40	40	40	40	40	40	40	40	40	40	18	14	04	14	03	00	@@@@@@@@.....
0090	07	24	07	00	08	14	74	00	05	24	0F	00	08	14	40	00	.\$...t.\$....@.
00A0	08	00	0E	11	47	D8	C4	C2	F2	61	C1	C9	E7	F6	F4	00	...G...a.....QDB2/AIX64.	
00B0	08	11	6D	94	81	A2	92	00	0C	11	5A	E2	D8	D3	F0	F9	..m.....Z.....	.._mask...]SQL09	
00C0	F0	F0	F0														...	000	

	ACCSEC	RQSDSS		(ASCII)	(EBCDIC)														
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF	0123456789ABCDEF	
0000	00	26	D0	00	10	00	20	00	10	6D	00	06	11	A2	00	03	..&.....	..m.....	..}.....S...
0010	00	16	21	10	E2	E3	D3	C5	C3	F1	40	40	40	40	40	40	..!.....@@@@@STLECI	
0020	40	40	40	40	40	40											@@@@@		

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 2 de 20)


```

3 data DB2 UDB DRDA Communication Manager sqljReceive 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

```

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 3 de 20)

```

4 data DB2 UDB DRDA Communication Manager sqljReceive fnc (3.3.54.3.0.1178)
  pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110549755 probe 1178
  bytes 122

```

RECEIVE BUFFER(AR):

	EXCSATRD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0059D04300010053 1443000F115EE5F8	.Y.C...S.C...^..	..}.....;V8
0010	F1C14BE2E3D3C5C3 F100181404140300	..K.....	1A.STLEC1.....
0020	0724070007147400 05240F0007144000	.\$...t..\$...@.
0030	0700081147D8C4C2 F20014116DE2E3D3	...G.....m...QDB2..._STL
0040	C5C3F14040404040 4040404040000C11	...@@@@@@@@@...	EC1 ...
0050	5AC4E2D5F0F8F0F1 F5	Z.....]DSN08015

	ACCSECRD OBJDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	0010D0030002000A 14AC000611A20003}.....s..

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 4 de 20)

```

5 data DB2 UDB DRDA Communication Manager sqljSend 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 11. Exemple de sortie de trace (connexion TCP/IP) (partie 5 de 20)

```
6 data DB2 UDB DRDA Communication Manager sqljcSend fnc (3.3.54.5.0.1177)
pid 807116 tid 1 cpid -1 node 0 sec 0 nsec 110659711 probe 1177
bytes 250
```

SEND BUFFER(AR):

	SECCHK RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	003CD04100010036 106E000611A20003	.<.A...6.n.....	..}.....>...s..
0010	00162110E2E3D3C5 C3F1404040404040	..!.....@#@#@@STLEC1
0020	404040404040000C 11A1D9858799F485	@#@#@@.....Regr4e
0030	A599000A11A09585 A6A39695	vr....newton

	ACCRDB RQSDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	00ADD001000200A7 20010006210F2407 !.\$.	..}....x.....
0010	00172135C7F9F1C1 F0C4F3C14BD7C1F8	..!5.....K...G91A0D3A.PA8
0020	F806030221064600 162110E2E3D3C5C3!.F.!.....	8.....STLEC
0030	F140404040404040 4040404040000C11	.@#@#@#@#@#@@... 1	...
0040	2EE2D8D3F0F9F0F0 F000D002FD8E3C4/... .SQL09000....QTD	
0050	E2D8D3C1E2C30016 00350006119C03335.....3	SQLASC.....
0060	0006119D04B00006 119E0333003C21043.	

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 6 de 20)

```
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

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 7 de 20)

```
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	000511A400u.

	ACCRDBRM RPYDSS	(ASCII)	(EBCDIC)
	0 1 2 3 4 5 6 7 8 9 A B C D E F	0123456789ABCDEF	0123456789ABCDEF
0000	009BD00200020095 2201000611490000".I..	..}....n.....
0010	000D002FD8E3C4E2 D8D3F3F7F0000C11	.../.....QTDSL370...
0020	2EC4E2D5F0F8F0F1 F5001600350006115...	.DSN08015.....
0030	9C04B80006119E04 B80006119D04B000
0040	0C11A0D5C5E6E3D6 D540400006212524@..!%\$...NEWTON
0050	34001E244E000624 4C00010014244D00	4..\$.N..\$.L...\$.M.+...<.....(.
0060	06244FFFFF000A11 E8091E768301BE00	.\$0.....v....	..!.....Y...c...
0070	2221030000000005 68B3B8C7F9F1C1F0	"!.....h.....G91A0
0080	C4F3C1D7C1F8F840 4040400603022106@#@@...!	D3APA88
0090	46000A11E8091E76 831389	F.....v....Y...c.i

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 8 de 20)

```

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 11. Exemple de sortie de trace (connexion TCP/IP) (partie 9 de 20)

```

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}.....

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 10 de 20)

```

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

```

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 11 de 20)

```

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\$..	..}.....

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 12 de 20)

```

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

```

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 13 de 20)

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	s1.mytable.	..._`./.%..

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 14 de 20)

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 11. Exemple de sortie de trace (connexion TCP/IP) (partie 15 de 20)

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 240800FFFFFFF3434	.f.....`\$.44	..}....-.....
0010	3237303444534E58 4F544C2000FFFFFFE	2704DSNXOTL+!.<.....
0020	0C00000000000000 00FFFFFFF000000
0030	0000000000572020 2057202020202020W W
0040	001053544C454331 2020202020202020	..STLEC1<.....
0050	2020000F44444353 5553312E4D595441	..DDCSUS1.MYTA(...
0060	424C450000FF	BLE...<.....

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 16 de 20)

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

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 17 de 20)

```
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}.....

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 18 de 20)

```
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

Figure 11. Exemple de sortie de trace (connexion TCP/IP) (partie 19 de 20)

```
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 11. Exemple de sortie de trace (connexion TCP/IP) (partie 20 de 20)

Concepts connexes :

- «Analyse du fichier de sortie de trace», à la page 126

Référence connexe :

- «Informations de mémoire tampon postérieures pour les traces DRDA», à la page 133

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** instruit au système de gestion du serveur de base de données hôte ou iSeries de valider l’unité d’oeuvre en cours. La quatrième mémoire tampon est reçue par le système de gestion du serveur de base de données hôte ou iSeries comme le résultat d’une validation ou d’une annulation. Il 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 DDM X'2408' suivi de X'FF'. Une SQLCA de valeur nulle (X'2408FF') indique le succès de l'opération (SQLCODE 0).

La figure 11, à la page 128 illustre un exemple d'une mémoire tampon de réception contenant une erreur SQLCA au niveau de trace 16.

Concepts connexes :

- «Analyse du fichier de sortie de trace», à la page 126

Référence connexe :

- «Modèles de fichiers de sortie de trace», à la page 128

Incidents DB2 Connect communs

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 un nombre différent de codes retour issus de DB2 Universal Database (UDB) pour iSeries, DB2 UDB pour OS/390 et z/OS et DB2 pour VM & 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 hôte ou iSeries ne peut être traduit. 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`, alors `db2diag.log` peut contenir une entrée similaire, par exemple :

```
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, svcname ou nname).

Un scénario possible consiste à posséder une connexion établie précédemment, 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 fructueuse 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

Lorsque vous vous êtes connecté à DB2 pour OS/390 et z/OS, les tables de bases de données de communications (CDB) n'ont pas été mises à jour correctement.

SQL30061 :

Symptôme

Connexion au mauvais emplacement de serveur de base de données hôte ou iSeries - 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 Universal Database pour z/OS et OS/390, 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) au lancement de l'utilitaire DDF (Distributed Data Facility).

Les commandes valides pour un noeud TCP/IP sont :

```
db2 catalog tcpip node <node_name> remote <host_name_or_address>
server <port_no_or_service_name>
db2 catalog dcs database <local_name> as <real_db_name>
db2 catalog database <local_name> as <alias> at node <node_name>
authentication server
```

Pour vous connecter à la base de données, exécutez la commande suivante :

```
db2 connect to <alias> user <user_name> using <password>
```

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. Elle peut également se produire lorsque vous vous connectez du serveur DB2 Connect vers un serveur de base de données hôte ou iSeries.

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 sqllib/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 DB2 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
```


Une fois DB2 arrêté ou redémarré, consultez le fichier db2diag.log afin de vérifier que les communications TCP/IP DB2 ont été démarrées. 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 "TCPIP" 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 et/ou le mot de passe sont
incorrects.
```

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 a été envoyé du client DB2 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.

Concepts connexes :

- «Identification d'incident», à la page 121
- «Utilitaire de trace», à la page 124

Référence connexe :

- «Erreurs de communication (message SQL30081N)» dans *Guide des messages - Volume 2*

Partie 3. Annexes

Annexe A. Transfert de données à l'aide de DB2 Connect

Si vous travaillez dans un environnement complexe qui nécessite un transfert de données entre un système de base de données hôte et un poste de travail, vous pouvez utiliser DB2 Connect, la passerelle du transfert de données entre l'hôte et le poste de travail (voir figure 12).

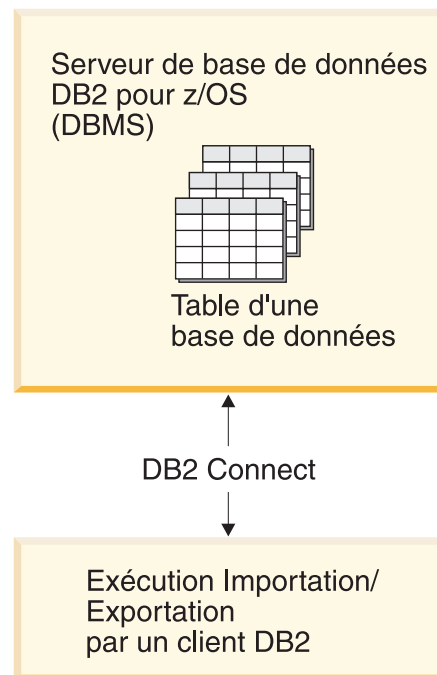


Figure 12. Importation et exportation via DB2 Connect

Les utilitaires d'exportation et d'importation DB2 permettent de transférer des données depuis une base de données située sur un serveur hôte ou iSeries vers un fichier situé sur le poste de travail DB2 Connect, et inversement. Par conséquent, vous pouvez utiliser les données avec toute autre application ou tout système de gestion de base de données relationnelle qui prend en charge ce format d'exportation ou d'importation. Par exemple, vous pouvez exporter des données d'une base de données située sur un serveur hôte ou iSeries vers un fichier PC/IXF puis l'importer dans une base de données DB2 pour Windows.

Vous pouvez exécuter des opérations d'exportation et d'importation depuis un client de base de données ou depuis un poste de travail DB2 Connect.

Remarques :

1. Les données à exporter ou importer doivent respecter les restrictions de taille et de type qui s'appliquent aux deux bases de données.
2. Vous pouvez optimiser les performances de l'importation en utilisant des requêtes composées. Indiquez le modificateur du type de fichier `multiple` dans l'utilitaire d'importation pour regrouper un nombre spécifique d'instructions de requête dans un bloc. Cette action permet de réduire le temps système du réseau et d'améliorer le temps de réponse.

Restrictions :

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

- Le fichier doit être de type PC/IXF.
- Vous devez créer une table cible avec des attributs compatibles avec les données sur le serveur cible avant de pouvoir l'importer. L'utilitaire **db2look** permet également d'extraire les attributs de la table source. L'importation via DB2 Connect ne permet pas de créer de table, puisque l'option INSERT est la seule prise en charge.

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

Remarque : Les définitions d'index ne sont ni stockées à l'exportation ni utilisées à l'importation.

Si vous exportez ou importez des données mixtes (des colonnes contenant à la fois des données à un octet et des données à deux octets), prenez soin de bien lire ce qui suit :

- Sur des systèmes qui stockent des données en EBCDIC (MVS, OS/390, OS/400, VM et VSE), les caractères hors code et les caractères de code normal indiquent le début et la fin des données à deux octets. Lors de la définition des longueurs de colonne de vos tables de base de données, vous devez attribuer suffisamment de place pour ces caractères.
- Nous vous recommandons d'utiliser des colonnes de caractères à longueur variable, excepté si les données de colonne disposent d'un modèle cohérent.

Transfert de données depuis un poste de travail vers un serveur hôte :

Pour transférer des données vers une base de données de serveur hôte ou AS/400 et iSeries :

1. Exportez les données 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 de la base de données du serveur hôte.

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

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

Exemple

L'exemple suivant illustre la procédure de transfert de données d'un poste de travail vers une base de données située sur un serveur hôte ou AS/400 et iSeries.

1. Exportez les données dans un format IXF externe en exécutant la commande suivante :

```
db2 export to staff.ixf of ixf select * from userid.staff
```
2. La commande suivante permet d'établir une connexion DRDA à la base de données DB2 cible :

```
db2 connect to cbc664 user admin using xxx
```
3. Le cas échéant, créez la table cible sur l'instance de base de données DB2 cible :

```
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))
```

4. Pour importer les données, exécutez la commande suivante :

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

Chaque ligne de données sera interprétée dans le fichier au format IXF et une instruction SQL INSERT sera émise pour insérer la ligne dans la table mydb.staff. Les lignes seront insérées une à une jusqu'à ce que l'ensemble des données soit transféré vers la table cible.

Vous trouverez des informations détaillées dans le guide de présentation suivant IBM Redbook: Moving Data Across the DB2 Family, disponible à l'adresse suivante : <http://www.redbooks.ibm.com/redbooks/SG246905.html>.

Concepts connexes :

- «Considérations sur le format de fichier lors du déplacement de données sur plusieurs plateformes» dans *Data Movement Utilities Guide and Reference*

Référence connexe :

- «Commande EXPORT» dans *Command Reference*
- «IMPORT (commande)» dans *Command Reference*

Annexe B. DB2 Database - Informations techniques

Présentation des informations techniques relatives à DB2

Les informations techniques relatives à DB2 sont disponibles via les méthodes et outils suivants :

- Centre de documentation DB2
 - Rubriques d'aide
 - Aide des outils DB2
 - Programmes exemples
 - Tutoriels
- Manuels DB2
 - Fichiers PDF (pouvant être téléchargés)
 - Fichiers PDF (à partir du CD DB2 PDF)
 - Documents imprimés
- Aide sur les lignes de commande
 - Aide sur les commandes
 - Aide sur les messages
- Programmes exemples

IBM met périodiquement à disposition des mises à jour de la documentation. Si vous accédez à la version en ligne du Centre de documentation DB2 sur le site ibm.com, il n'est pas nécessaire d'installer les mises à jour de la documentation car cette version est maintenue à jour par IBM. Si vous avez installé le Centre de documentation DB2, il est recommandé d'installer les mises à jour de la documentation. Les mises à jour de la documentation permettent d'actualiser les informations installées à partir du *CD du Centre de documentation DB2* ou téléchargées sur Passport Advantage au fur et à mesure qu'elles sont mises à disposition.

Remarque : Les rubriques du Centre de documentation DB2 sont mises à jour plus souvent que les guides papier ou les versions PDF. Pour obtenir les informations les plus récentes, installez les mises à jour de la documentation dès qu'elles deviennent disponibles ou consultez le Centre de documentation DB2 sur le site ibm.com.

Vous pouvez accéder en ligne à des informations techniques complémentaires concernant DB2 comme par exemple les notes techniques, les livres blancs ou les Redbooks, sur le site ibm.com. Accédez au site DB2 Information Management software library à l'adresse <http://www.ibm.com/software/data/sw-library/>.

Commentaires sur la documentation

Vos commentaires sur la documentation DB2 sont les bienvenus. Pour toutes vos suggestions d'amélioration de la documentation DB2, envoyez un courrier électronique à l'adresse suivante : db2docs@ca.ibm.com. L'équipe chargée de la documentation DB2 lit tous vos commentaires, mais ne peut pas vous répondre directement. Donnez des exemples spécifiques chaque fois que possible afin que

nous puissions mieux comprendre vos problèmes. Si vous nous envoyez un commentaire sur une rubrique ou un fichier d'aide spécifique, n'oubliez pas de mentionner son titre ou son adresse URL.

N'utilisez pas cette adresse électronique pour contacter le Service Clients DB2. Si vous ne pouvez pas résoudre un incident technique lié à DB2 à l'aide de la documentation, contactez le service d'assistance IBM local.

Concepts connexes :

- «Fonctions du Centre de documentation DB2» dans le *Centre de documentation DB2 en ligne*
- «Fichiers modèles» dans les *exemples de Rubriques*

Tâches connexes :

- «Appel de l'aide sur les commandes à partir de l'interpréteur de commandes» dans *Command Reference*
- «Appel de l'aide sur les messages à partir de l'interpréteur de commandes» dans *Command Reference*
- «Mise à jour du Centre de documentation installé sur votre ordinateur ou sur un serveur intranet», à la page 151

Référence connexe :

- «Bibliothèque technique DB2 au format PDF», à la page 146

Bibliothèque technique DB2 au format PDF

Le tableau suivant décrit la bibliothèque DB2 disponible dans le centre de publications IBM à l'adresse suivante www.ibm.com/shop/publications/order.

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.

Les informations contenues dans ces manuels sont très utiles pour l'ensemble des utilisateurs DB2, que vous soyez programmeur ou administrateur de base de données ou que vous travailliez avec DB2 Connect ou d'autres produits DB2.

Tableau 16. Informations techniques sur DB2

Nom	Référence	Disponible au format papier
<i>Administration Guide: Implementation</i>	SC10-4221	Oui
<i>Administration Guide: Planning</i>	SC10-4223	Oui
<i>Administrative API Reference</i>	SC10-4231	Oui
<i>Routines et vues SQL d'administration</i>	SC10-4293	Non
<i>Call Level Interface Guide and Reference, Volume 1</i>	SC10-4224	Oui
<i>Call Level Interface Guide and Reference, Volume 2</i>	SC10-4225	Oui
<i>Command Reference</i>	SC10-4226	Non

Tableau 16. Informations techniques sur DB2 (suite)

Nom	Référence	Disponible au format papier
<i>Data Movement Utilities Guide and Reference</i>	SC10-4227	Oui
<i>Data Recovery and High Availability Guide and Reference</i>	SC10-4228	Oui
<i>Developing ADO.NET and OLE DB Applications</i>	SC10-4230	Oui
<i>Developing Embedded SQL Applications</i>	SC10-4232	Oui
<i>Developing SQL and External Routines</i>	SC10-4373	Non
<i>Developing Java Applications</i>	SC10-4233	Oui
<i>Developing Perl and PHP Applications</i>	SC10-4234	Non
<i>Getting Started with Database Application Development</i>	SC10-4252	Oui
<i>Guide d'initiation à l'installation et à l'administration de DB2 sous Linux et Windows</i>	GC11-2423	Oui
<i>Guide des messages, volume 1</i>	SC11-2428	Non
<i>Guide des messages, volume 2</i>	SC11-2429	Non
<i>Guide de migration</i>	GC11-2424	Oui
<i>Net Search Extender Administration and User's Guide</i>	SH12-6842	Oui
Remarque : La version HTML de ce document ne figure pas sur le CD-ROM de documentation HTML.		
<i>Performance Guide</i>	SC10-4222	Oui
<i>Query Patroller - Guide d'utilisation et d'administration</i>	GC11-2421-00	Oui
<i>DB2 Clients - Guide d'initiation</i>	GC11-2425	Non
<i>DB2 Servers - Guide d'initiation</i>	GC11-2422	Oui
<i>Spatial Extender and Geodetic Data Management Feature User's Guide and Reference</i>	SC18-9749	Oui
<i>SQL Guide</i>	SC10-4248	Oui
<i>SQL Reference, Volume 1</i>	SC10-4249	Oui
<i>SQL Reference, Volume 2</i>	SC10-4250	Oui
<i>System Monitor Guide and Reference</i>	SC10-4251	Oui
<i>Troubleshooting Guide</i>	GC10-4240	Non
<i>Visual Explain Tutorial</i>	SC11-2643	Non
<i>Nouveautés</i>	SC11-2430	Oui
<i>XML Extender Administration and Programming</i>	SC18-9750	Oui

Tableau 16. Informations techniques sur DB2 (suite)

Nom	Référence	Disponible au format papier
<i>Guide XML</i>	SC10-4254	Oui
<i>XQuery Reference</i>	SC18-9796	Oui

Tableau 17. Informations techniques spécifiques à DB2 Connect

Nom	Référence	Disponible au format papier
<i>DB2 Connect - Guide de l'utilisateur</i>	SC11-2427	Oui
<i>DB2 Connect Personal Edition - Guide d'initiation</i>	GC11-2561	Oui
<i>DB2 Connect Servers - Guide d'initiation</i>	GC11-2426	Oui

Tableau 18. Informations techniques sur WebSphere Information Integration

Nom	Référence	Disponible au format papier
<i>WebSphere Information Integration: Administration Guide for Federated Systems</i>	SC19-1001	Oui
<i>WebSphere Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i>	SC19-1000	Oui
<i>WebSphere Information Integration: Configuration Guide for Federated Data Sources</i>	SC19-1034	Non
<i>WebSphere Information Integration: SQL Replication Guide and Reference</i>	SC19-1002	Oui

Remarque : Les notes sur l'édition DB2 contiennent des informations supplémentaires concernant la version de votre produit et le niveau du groupe de correctifs. Pour plus d'informations, consultez les liens correspondants.

Concepts connexes :

- «Présentation des informations techniques relatives à DB2», à la page 145
- «A propos des Notes sur l'édition» dans les *Notes sur l'édition*

Tâches connexes :

- «Commande de manuels imprimés DB2», à la page 149

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 en commander auprès de votre représentant IBM. Gardez à l'esprit que certains manuels au format électronique sur le CD 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 CD 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 CD-ROM 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/db2help/> .

Procédure :

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 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 :
 - 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".
 - Si vous appelez, précisez que vous souhaitez commander une publication DB2.
 - Indiquez à votre agent les titres et les numéros de référence des manuels que vous souhaitez commander.

Concepts connexes :

- «Présentation des informations techniques relatives à DB2», à la page 145

Référence connexe :

- «Bibliothèque technique DB2 au format PDF», à la page 146

Affichage de l'aide sur les codes d'état SQL à partir de l'interpréteur de commandes

DB2 renvoie une valeur SQLSTATE pour les conditions qui pourraient 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.

Procédure :

Pour accéder à 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.

Tâches connexes :

- «Appel de l'aide sur les commandes à partir de l'interpréteur de commandes» dans *Command Reference*
- «Appel de l'aide sur les messages à partir de l'interpréteur de commandes» dans *Command Reference*

Accès aux différentes versions du Centre de documentation DB2

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 <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>.

Tâches connexes :

- «Configuration de l'accès à l'aide contextuelle et à la documentation de DB2» dans *Administration Guide: Implementation*

Affichage des rubriques du Centre de documentation DB2 dans votre langue préférée

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.

Procédure :

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...**

Remarque : L'ajout d'une langue ne garantit pas que l'ordinateur dispose des polices requises pour afficher les rubriques dans votre langue préférée.

- 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. Effacez le contenu de la mémoire cache du navigateur, puis régénérez la page pour afficher le Centre de documentation DB2 dans votre langue préférée.

Pour afficher les rubriques dans votre langue préférée dans un navigateur Firefox ou Mozilla, procédez comme suit :

1. Sélectionnez **Outils** —> **Options** —> **Langues**. Le panneau Langues s'affiche 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. Effacez le contenu de la mémoire cache du navigateur, puis régénérez la page pour afficher le Centre de documentation DB2 dans votre langue préférée.

Pour certaines combinaisons de navigateur et de système d'exploitation, il se peut que vous deviez également modifier les paramètres régionaux de votre système d'exploitation pour spécifier l'environnement local et la langue de votre choix.

Concepts connexes :

- «Présentation des informations techniques relatives à DB2», à la page 145

Mise à jour du Centre de documentation installé sur votre ordinateur ou sur un serveur intranet

Si un Centre de documentation DB2 est installé localement, les rubriques mises à jour peuvent être disponibles pour téléchargement. La mention 'Dernière mise à jour' se trouvant au bas de la plupart des rubriques indique le niveau actuel de la rubrique.

Pour déterminer si une mise à jour est disponible pour l'intégralité du Centre de documentation DB2, recherchez la valeur 'Dernière mise à jour' sur la page d'accueil du Centre de documentation. Comparez la valeur de votre page d'accueil installée localement à celle qui est disponible sur la page d'accueil du Centre de documentation hébergé par IBM. Si elles sont différentes, vous disposez du dernier niveau de documentation et aucune mise à jour n'est requise. Dans le cas contraire, vous devez mettre à jour le Centre de documentation installé localement.

Pour la mise à jour du Centre de documentation DB2 installé localement, vous devez effectuer les actions suivantes :

1. Arrêtez le Centre de documentation DB2 sur votre ordinateur et redémarrez le Centre de documentation en mode autonome. L'exécution du Centre de documentation en mode autonome empêche les autres utilisateurs du réseau d'accéder au Centre de documentation et permet de télécharger et d'appliquer les mises à jour.

2. Utilisez la fonction de mise à jour afin de déterminer si des modules de mise à jour sont disponibles à partir d'IBM. Le cas échéant, utilisez la fonction de mise à jour pour télécharger les modules. (La fonction de mise à jour est disponible uniquement en mode autonome.)
3. Arrêtez le Centre de documentation autonome et redémarrez le Centre de documentation DB2 sur votre ordinateur.

Procédure :

Pour mettre à jour le Centre de documentation DB2 installé sur votre ordinateur ou le serveur intranet, procédez comme suit :

1. Arrêtez le service Centre de documentation DB2.
 - Sous Windows, cliquez sur **Démarrer** → **Panneau de configuration** → **Outils d'administration** → **Services**. Puis cliquez à l'aide du bouton droit de la souris sur le service Centre de documentation **DB2** et sélectionnez **Arrêter**.
 - Sous Linux, entrez la commande suivante :


```
/etc/init.d/db2icdv9 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 se trouve dans le répertoire C:\Program Files\IBM\DB2 Information Center\Version 9.
 - c. Exécutez le fichier help_start.bat à partir du chemin complet du Centre de documentation DB2 :


```
<Rép Centre de documentation DB2 >\doc\bin\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/V9.
 - b. Exécutez le fichier help_start.sh à partir du chemin complet du Centre de documentation DB2 :


```
<Rép  
Centre de documentation DB2 >/doc/bin/help_start
```

Le navigateur Web par défaut du système affiche le Centre de documentation autonome.
3. Cliquez sur le bouton Mise à jour (🔄). Sur le côté droit du panneau du Centre de documentation, cliquez sur l'option de recherche de mises à jour. Une liste des mises à jour des documentations existantes s'affiche.
4. Pour lancer le processus de mise à jour, sélectionnez les éléments à télécharger puis cliquez sur l'option d'installation des mises à jour.
5. Une fois le processus de téléchargement et d'installation terminé, cliquez sur **Terminer**.
6. Arrêtez le Centre de documentation autonome.
 - Sous Windows, exécutez le fichier help_end.bat à partir du chemin complet du Centre de documentation DB2 :


```
<Rép Centre de documentation DB2  
>\doc\bin\help_end.bat
```
 - Sous Linux, exécutez le fichier help_end.sh à partir du chemin complet du Centre de documentation DB2 :

<Rép Centre de documentation DB2 >/doc/bin/help_end

7. Redémarrez le service Centre de documentation DB2.

- Sous Windows, cliquez sur **Démarrer** → **Panneau de configuration** → **Outils d'administration** → **Services**. Puis cliquez à l'aide du bouton droit de la souris sur le service Centre de documentation **DB2** et sélectionnez **Démarrer**.

- Sous Linux, entrez la commande suivante :

```
/etc/init.d/db2icdv9 start
```

Le Centre de documentation DB2 mis à jour affiche les nouvelles rubriques et les rubriques mises à jour.

Concepts connexes :

- «Options d'installation du Centre de documentation DB2» dans *Serveurs DB2 - Mise en route*

Tâches connexes :

- «Installation du Centre de documentation DB2 via l'assistant d'installation DB2 (Linux)» dans *Serveurs DB2 - Mise en route*
- «Installation du Centre de documentation DB2 via l'assistant d'installation DB2 (Windows)» dans *Serveurs DB2 - Mise en route*

Tutoriel DB2 Visual Explain

Le tutoriel DB2 Visual Explain vous permet d'apprendre à analyser, optimiser et régler les instructions SQL pour améliorer les performances. 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.

Tutoriel DB2 Visual Explain :

Pour afficher le tutoriel, cliquez sur le titre.

Tutoriel Visual Explain

Analyse, optimisation et ajustement des instructions SQL pour l'optimisation des performances à l'aide de Visual Explain.

Concepts connexes :

- «Présentation de Visual Explain» dans *Administration Guide: Implementation*

Informations concernant la résolution d'incidents liés à DB2

Un grand nombre d'informations concernant l'identification et la résolution d'incidents sont à votre disposition lorsque vous utilisez les produits DB2.

Documentation DB2

Les informations relatives à l'identification des incidents sont disponibles dans le document DB2 Troubleshooting Guide ou dans la section Support and Troubleshooting du Centre de documentation DB2. Vous y trouverez des informations sur la manière d'isoler et d'identifier les incidents à l'aide des utilitaires et outils de diagnostic de DB2, des solutions aux incidents les plus courants et d'autres conseils sur la résolution des incidents que vous pourriez rencontrer avec vos produits 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/udb/support.html>

Concepts connexes :

- «Introduction à l'identification d'incident» dans *Troubleshooting Guide*
- «Présentation des informations techniques relatives à DB2», à la page 145

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 susmentionnées 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 C. 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 EMEA Director of Licensing
IBM Europe Middle-East Africa
Tour Descartes
La Défense 5
2, avenue Gambetta
92066 - Paris-La Défense CEDEX
France

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 :

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japon

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. Il 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.

Ce document peut contenir des exemples de données et des 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 peut contenir des exemples de programmes 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 lesquelles 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.

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. Programmes exemples. © Copyright IBM Corp. *_indiquez l'année ou les années_*. All rights reserved.

Marques

Les noms de sociétés, de produits et de services apparaissant dans les documents de la bibliothèque de documentation DB2 peuvent appartenir à International Business Machines Corporation ou à des tiers. Des informations sur les marques d'IBM Corporation aux Etats-Unis et/ou dans certains autres pays sont disponibles sur le site <http://www.ibm.com/legal/copytrade.shtml>.

Les termes qui suivent sont des marques d'autres sociétés et sont utilisés dans l'un des documents de la bibliothèque de documentation DB2 :

Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

Intel, Itanium, Pentium et Xeon sont des marques de Intel Corporation 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.

Linux est une marque de Linus Torvalds 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

Caractères spéciaux

- „ (virgule virgule) dans une chaîne de paramètres 35
- , (virgule) dans une chaîne de paramètres 35

A

- accès direct à une base de données
 - DB2 Connect PE 17
- ACCRDB (commande) 126
- ACCRDBRM (commande) 126
- ACCSEC (commande) 126
- AGENTPRI (paramètre) 107
- aide
 - affichage 150
 - pour les instructions SQL 150
- alias de base de données client 77
- améliorations apportées par chaque nouvelle édition 4
- applications
 - définition des accès 57
 - performances 92
 - procédures mémorisées 92
 - SQL composé 92
 - Web
 - au moyen de DB2 Connect 20
- applications Web
 - DB2 Connect Enterprise Edition 20
 - procédures mémorisées 23
- architecture CDRA (Character Data Representation Architecture) 12
- Architecture de base de données relationnelle répartie (DRDA)
 - accès aux données 11
 - architecture CDRA (Character Data Representation Architecture) 12
 - architectures 12
 - concepts 11
 - demandeur d'application 12
 - Distributed Data Management (DDM) 12
 - flot de données 12
 - Formatted Data Object Content Architecture (FDOCA) 12
 - Management Services Architecture (MSA) 12
 - présentation 11
 - serveur d'applications 12
 - TCPIP 12
- assistants
 - mise à jour multisite 62
- attributs du serveur d'échanges de données (commande) 126
- authentification 40
 - présentation 45
 - types
 - CLIENT 45, 53
 - DCE 45
 - KERBEROS 45
 - SERVER 45
 - SERVER_ENCRYPT 45
 - valeur par défaut 45
 - validation 45
- automatic client reroute
 - configuration 84
 - description 84
 - échecs de connexion 86

B

- bases de données
 - alias 33, 40
 - concepts
 - MVS 6
 - OS/390 6
 - OS/400 6
 - VM 6
 - VSE 6
 - z/OS 6
 - nom 33, 35, 40
 - objet RDBNAM 126
 - optimisation 109
 - outils de mesure des performances 89
 - regroupement de requêtes 92
- bases de données cible
 - nom 35, 40
- bases de données fédérées
 - requête répartie 15
- BINDADD (droit)
 - droits requis pour la définition des accès 57
- bloc de requête supplémentaire
 - CLI/ODBC 116
 - JDBC 116
 - SQL imbriqué 116
- blocs de requêtes, augmentation de la vitesse de transfert des données DB2 Connect 115

C

- caractères d'échappement 42
- CCSID (ID de jeu de caractères codés)
 - support des CCSID bidirectionnels
 - description 35
- Centre de contrôle
 - mises à jour multisites 62
- Centre de documentation
 - affichage dans différentes langues 150
 - mise à jour 151
 - versions 150
- Centre de documentation DB2
 - affichage dans différentes langues 150
 - mise à jour 151
 - versions 150
- changement d'échelle des fenêtres, extensions RFC-1323 117
- CHAR (type de données)
 - description 118
- CLI
 - connexions sécurisées 47
- CLI (call level interface)
 - applications
 - CURRENTPACKAGESET 53
 - présentation 114
- CLIENT (type d'authentification)
 - remarques relatives à DB2 Connect 45
- client reroute
 - automatiques 84
- commande commit
 - mémoires tampon de sortie de la trace 126

- commandes
 - ACCRDB 126
 - ACCRDBRM 126
 - ACCSEC 126
 - commit 126
 - EXCSAT 126
 - EXCSATRD 126
 - GET SNAPSHOT 75
 - SECCHK 126
- commandes de manuels DB2 149
- COMMIT (instruction)
 - liaison statique 92
- concentrateur XA, exemples 98
- concentrateurs de connexions
 - agents 98
 - agents logiques 98
 - comparaison avec le regroupement de connexions 103
 - exemples 98
 - MAX_COORDAGENTS (paramètre de configuration) 98
 - MAXAGENTS (paramètre de configuration) 98
 - mise en oeuvre 98
 - NUM_INITAGENTS (paramètre de configuration) 98
 - NUM_POOLAGENTS (paramètre de configuration) 98
 - paramètres de configuration 98
 - présentation 95
 - regroupement 98
 - répartiteur 98
 - restrictions 98
 - support des transactions XA 98
 - temps système 98
- conception d'applications 92
- conditionnement du produit 3
- configuration
 - connexions hôte 17
 - remarques, changement du mot de passe 53
- Configure Multisite Update wizard 62
- conflit
 - ressources système 111
- connectivité
 - serveurs, DB2 Connect Enterprise Edition 19
- connectivité à la base de données hôte
 - équilibrage de la charge 83
 - haut niveau de disponibilité 83
- connexions
 - concentrateurs, voir concentrateurs de connexions 98
 - connexion directe à un hôte 17
 - DB2 Connect Enterprise Edition 19
 - reestablishing
 - connexion directe à un hôte 17
 - DB2 Connect Enterprise Edition 19
 - regroupement
 - avantages 98
 - concentrateurs de connexions 98
 - présentation 95
- connexions sécurisées 47
 - changement d'utilisateurs via CLI/ODBC 50
 - via CLI/ODBC 49
- contacter IBM 163
- contexte sécurisé
 - support DB2 Connect 47
 - via CLI/ODBC 49
- contrôle
 - connexions
 - Serveur DB2 Connect 73
 - moniteur de performances Windows 74
- conversions
 - données hôte 118

- CREATE IN COLLECTION NULLID (droit) 57
- CURRENTPACKAGESET (mot-clé CLI/ODBC) 53

D

- dans le fichier de mappage des SQLCODE 67
- dates
 - support des décalages horaires 35
- DB2 Connect
 - améliorations apportées aux versions précédentes 4
 - concepts 9
 - DCE
 - sécurité 53
 - présentation 3
 - scénarios
 - moniteurs de traitement des transactions 17
 - support de l'environnement Sysplex 103
- DB2 Connect Enterprise Edition
 - applications Web 20
 - gestionnaire de transactions compatible XA 64
 - Interfaces API 22
 - JDBC 22
 - moniteurs de traitement des transactions 27
 - serveur de connectivité 19
 - serveur de connectivité, scénarios 17
 - serveurs Web 23
 - SQLJ 22
 - Tuxedo 27
- DB2 Connect Personal Edition
 - description du produit 3
- DB2 Universal Database pour OS/390 et z/OS 34
 - améliorations apportées à la sécurité
 - codes de sécurité étendus 53
 - sécurité des applications Java et ODBC de bureau 53
 - sécurité TCP/IP déjà vérifiée (option) 53
 - support du changement des mots de passe 53
 - DOMAIN 34
 - DYNAMICRULES(BIND) 53
 - fichier d'amorçage 34
 - paramètres BSDS 34
 - RESPORT 34
 - TCPPORT 34
- db2drdat (utilitaire)
 - fichier de sortie 124
- DCE
 - conditions requises 53
 - type d'authentification 45
- dcs1ari.map (fichier) 67
- dcs1dsn.map (fichier) 67
- dcs1qsq.map (fichier) 67
- ddcs400.lst (fichier) 57
- ddcsmvs.lst (fichier) 57
- ddcstrc (utilitaire)
 - fichier de sortie 125
- ddcsvm.lst (fichier) 57
- ddcsvse.lst (fichier) 57
- débit
 - transactions 89
- définition des accès
 - droits
 - marqueurs de paramètres avec décalage 57
 - modules 57
 - noms de module 57
 - utilitaires et applications 57
- demandeurs d'applications
 - définition de l'architecture DRDA 12
 - paramètres 40

- DESCRIBE (instruction) 92
- développement d'applications 92
 - client DB2 AD 17
 - ODBC 17
- DIRCACHE (paramètre) 107
- Distributed Data Management (DDM) 12, 124
- documentation 145, 146
 - modalités d'utilisation 154
- Documents imprimés
 - tri 149
- données
 - conversions
 - hôte 118
 - flots 12
 - performances 89
 - groupage 92
 - performances pour le transfert 119
 - sources
 - requête répartie 15
 - vitesse de transfert 89, 119
- données de type caractère 118
- données décimales non condensées 118
- droits d'accès
 - définition des accès 57
- DSS (sous-section répartie)
 - type, trace 124

E

- échecs de connexion
 - réacheminement automatique des clients 86
- erreurs
 - identification des incidents 121
- erreurs de transmission client 84
- état du système, commande GET SNAPSHOT 75
- EXCSAT (commande) 126
- EXCSATRD (commande) 126
- EXECUTE IMMEDIATE (instruction)
 - conception d'applications 92
- exemples
 - concentrateurs de connexions 98
 - concentrateurs XA 98
- EXTNAM (objet) 126
- EXTRA BLOCKS SRV (paramètre) 116

F

- feuilles de travail
 - personnalisation des répertoires 40
- fichiers core
 - identification des incidents 122
- fonction de contrôle d'accès aux données (RACF)
 - sécurité 54
- fonction de trace
 - buffer information for DRDA traces 133
 - output file samples 128
- FOR FETCH ONLY (clause)
 - SELECT (instruction) 92
- FORCE (commande)
 - ID agent 77
- format décimal condensé 118
- Formatted Data Object Content Architecture (FDOCA) 12
- fuseaux horaires 35

G

- gestionnaire de points de synchronisation (SPM)
 - paramètres par défaut 64
 - scénarios 63
- gestionnaires de ressources XA 27
- gestionnaires de transactions XA
 - concentrateurs de connexions 98
 - description 27
- GET SNAPSHOT (commandes) 75
- goulots d'étranglement
 - performances 89
 - transactions 89
- GRANT (instruction)
 - sécurité 54
- groupage de données 92

I

- IBM SQL 7
- IBM WebSphere 21
- ID application client 77
- ID application hôte 77
- ID autorisation 77
- ID de page de codes 77
- ID produit client 77
- ID produit hôte 77
- identification des incidents
 - collecte des informations 121
 - connect 122, 123
 - DB2 Connect 134
 - fonctions de trace
 - DRDA 128, 133
 - informations en ligne 153
 - outils de diagnostic 122
 - performances 112
 - présentation 121
 - problèmes liés à la connexion 122
 - problèmes postérieurs à la connexion 123
 - tutoriels 153
- instructions
 - COMMIT 92
 - DESCRIBE 92
 - EXECUTE IMMEDIATE 92
 - FOR FETCH ONLY 92
 - PREPARE 92
 - ROLLBACK
 - conception d'applications 92
 - SELECT 92
- instructions SQL
 - affichage de l'aide 150
- INTEGER
 - type de données 118
- interpréteur de commandes
 - instructions SQL 8
 - performances 92
- INTERRUPT_ENABLED (déconnexion), paramètre 35
- iSeries
 - DRDA 12

J

- Java
 - serveurs d'applications
 - DB2 Connect EE 22
 - Interfaces API 22
 - JDBC 22

Java (*suite*)
 serveurs d'applications (*suite*)
 SQLJ 22
jetons
 SQLCODE 67
journal de diagnostic de premier niveau 122

K

Kerberos
 type d'authentification 45
 pour OS/390 46
 sur z/OS 46

L

LIST DCS APPLICATIONS (commande) 77
liste d'adresses placées dans la mémoire cache 105
liste des fichiers de liens 57
LOCALDATE (paramètre) 35

M

mappage des codes SQLCODE 67
 adaptation 67
 NOMAP (paramètre) 67
matériel
 performances du réseau 119
MAX_COORDAGENTS (paramètre) 95, 98
MAXAGENTS (paramètre) 98, 107
MAXDARI configuration parameter 107
maxRetriesForClientReroute 84
mémoire tampon d'envoi, fonction de trace 124
message de réponse indiquant la fin d'une unité d'oeuvre
 (ENDUOWRM) 126
messages d'erreurs
 DB2 Connect 134
Microsoft Windows
 applications 17
mise en mémoire cache des répertoires (paramètre de
 configuration)
 Optimisation de DB2 Connect 107
mises à jour
 Centre de documentation 151
 Centre de documentation DB2 151
 répertoires de base de données 33
mises à jour multisites
 activation 61
 Centre de contrôle 62
 gestionnaire de points de synchronisation 63
 test 63
 unité d'oeuvre répartie (DUOW) 61
modalités
 utilisation des publications 154
modèle DTP (Distributed Transaction Processing) X/Open 27
modules
 créés sur un serveur de base de données hôte ou
 iSeries 57
moniteur du gestionnaire de bases de données
 clients éloignés 73
 description 8
moniteurs de traitement des transactions
 caractéristiques 27
 exemples 27
 mises à jour multisites 61
 OLTP 27

moniteurs de traitement des transactions (*suite*)
 transactions 27
 Tuxedo 27
mots de passe
 changement (OS/390 and z/OS) 53

N

NNAME client 77
noeud SOCKS
 variables d'environnement obligatoires 34
noeuds
 nom 33, 34, 40
 répertoire 33, 34
nom de base de données cible AS 35
nom de destination symbolique 40
 distinction majuscules/minuscules 34
nom de l'application (élément de contrôle) 77
nom de la base de données hôte (élément de contrôle) 77
NOMAP (paramètre) 35, 67
NONE (types de sécurité) 54
NULLID
 OS/400 57
NUM_INITAGENTS (paramètre) 95, 98
NUM_POOLAGENTS (paramètre) 95, 98
NUMDB (paramètre) 107
numéro de séquence client 77
numéro de séquence sortante 77

O

objet SRVNAM 126
ODBC (Open Database Connectivity)
 applications
 CURRENTPACKAGESET 53
 interface 17
 optimisation de l'accès 113
 présentation 114
optimisation
 DB2 pour OS/390 et z/OS 112
 DIRCACHE (paramètre) 107
 MAXAGENTS (paramètre) 107
 MAXDARI (paramètre) 107
 NUMDB (paramètre) 107
 performances
 base de données 109
 réseau 109
 RQRIOBLK (paramètre) 107
OS/390
 DRDA 12
OS/400
 DRDA 12
outils
 diagnostic 122
 performances 89
 Utilisation de l'UC 89
 utilisation de la mémoire 89
outils d'exploitation de l'UC 89
outils d'exploitation de la mémoire 89
outils de diagnostic
 identification des incidents 122

P

paramètre D (déconnexion) 35

- paramètres
 - AGENTPRI 107
 - BIDI 35
 - chaînes 42
 - D (déconnexion) 35
 - DIRCACHE 107
 - EXTRA BLOCKS SRV 116
 - INTERRUPT_ENABLED (déconnexion) 35
 - LOCALDATE 35
 - MAX_COORDAGENTS 98
 - MAXAGENTS 98, 107
 - MAXDARI 107
 - NOMAP 35
 - NUM_INITAGENTS 98
 - NUM_POOLAGENTS 98
 - NUMDB 107
 - paramètres de répertoire 40
 - PRDID 126
 - RQRIOBLK 107
 - SYSLEX 35
 - virgule dans les chaînes 35
- paramètres de configuration
 - MAX_COORDAGENTS 95
 - NUM_INITAGENTS 95
 - NUM_POOLAGENTS 95
 - TCP_KEEPALIVE 86
- paramètres de fichiers d'amorçage BSDS (Bootstrap Data Set)
 - z/OS et OS/390 34
- performances
 - applications
 - conception 92
 - groupage de données 92
 - logique des prédicats 92
 - procédures mémorisées 92
 - regroupement de requêtes 92
 - SQL composé 92
 - concentrateur de connexion 103
 - concepts 89
 - DB2 pour OS/390 et z/OS 112
 - flots de données 89
 - goulots d'étranglement 89
 - identification des incidents 112
 - increasing transfer rates 115
 - interpréteur de commandes (CLP) 92
 - matériel de réseau 119
 - mesures 89
 - optimisation 112
 - optimizing ODBC access 113
 - outils 89
 - outils de réseau 89
 - regroupement de connexions 103
 - remarques relatives à SQL 92
 - ressources système 111
 - tests de performances 89
- perluète (double ())
 - fichier de mappage des SQLCODE 67
- personnalisation
 - répertoires, feuilles de travail associées 40
- pilote JDBC DB2 Universal
 - client reroute support 84
- PRDID (paramètre) 126
- prédicats
 - performances de la logique 92
- PRÉPARE (instruction)
 - dans la conception d'une application 92
 - effet sur les performances 92

- présentation
 - DB2 Connect 3
- pris en charge
 - transactions XA 98
- procédures mémorisées
 - présentation 23
- produits serveur DB2 Connect
 - description du produit 3
- programmation CGI (Common Gateway Interface, Interface de passerelle commune)
 - avantages 20
 - restrictions 20

R

- receive buffer 124
- références
 - defining multiple database entries 41
- regroupement de connexions 95
 - comparaison avec le concentrateur de connexion 103
 - présentation 95
- regroupement des requêtes concernant les bases de données
 - performances 92
- Relational Connect
 - description du produit 9
- relations de confiance
 - contextes et connexions sécurisés 47
- remarques 157
- répertoire DCS
 - BIDI (paramètre) 35
 - contenu 35
 - LOCALDATE (paramètre) 35
 - nom de base de données 35
 - nom de base de données cible AS 35
 - nom de la base de données cible 35
 - spécification de la chaîne de paramètres 42
 - SYSLEX (paramètre) 35
- répertoire DCS (Database Connection Services)
 - mise à jour d'entrées 33
- répertoire système des bases de données
 - alias de base de données 33
 - authentification 33
 - avant la mise à jour 33
 - nom de base de données 33
 - nom de noeud 33
 - valeurs 33
- répertoires
 - personnalisation
 - feuilles de travail 40
- répertoires de base de données
 - base de données système 33
 - Database Connection Services (DCS) 33
 - entrées multiples 41
 - mise à jour 33
 - noeud 33
- requêtes réparties
 - bases de données fédérées 15
 - compensation 15
 - définition 15
 - support 15
 - transparence des emplacements 15
- réseau
 - data transfer rates 119
 - optimisation 109
 - outils de mesure des performances 89
- ressources système, conflit 111

- restrictions
 - concentrateur de connexion 98
- retryIntervalForClientReroute 84
- REVOKE (instruction)
 - sécurité 54
- ROLLBACK (instruction)
 - liaison statique 92
- RQRIOLBK (paramètre)
 - optimisation 107

S

- SAME (type de sécurité) 54
- scénarios
 - sécurité TCP/IP 54
- SECCHK (commande) 126
- sécurité
 - codes étendus
 - OS/390 et z/OS 53
 - conseils 53
 - DB2 Connect
 - remarques 53
 - support 54
 - GRANT (instruction) 54
 - Kerberos
 - XXXX 46
 - REVOKE (instruction) 54
 - TCP/IP 54
 - types 40
 - valeurs du répertoire des noeuds 34
- SELECT (instruction)
 - dans la conception d'une application 92
 - FOR FETCH ONLY 92
 - modifiables 92
- SERVER (type d'authentification) 45
- SERVER_ENCRYPT (type d'authentification) 45
- serveurs
 - alternate 84
 - application
 - DB2 Connect EE 24
- serveurs d'applications
 - clients lourds 24
 - configuration 24
 - DB2 Connect ESE 24
 - définition de l'architecture DRDA 12
 - déploiement 24
 - modèle à deux niveaux 24
 - modèle à trois niveaux 24
 - présentation 24
 - support DB2 Connect 24
- serveurs Web
 - DB2 Connect Enterprise Edition 23
- SET CURRENT PACKAGESET (instruction) 53
- SHOW DETAIL (option du moniteur) 77
- SQL (Structured Query Language)
 - dynamique 92
 - statique 92
- SQL_ATTR_
 - TRUSTED_CONTEXT_PASSWORD
 - utilisation 50
 - TRUSTED_CONTEXT_USERID
 - utilisation 50
 - USE_TRUSTED_CONTEXT
 - utilisation 49
- SQL composé
 - NON ATOMIQUE 92

- SQL composé ATOMIQUE
 - non pris en charge dans DB2 Connect 92
- SQL composé NON ATOMIQUE
 - conception d'applications 92
- SQL/DS
 - DRDA 12
- SQL dynamique
 - CURRENTPACKAGESET 53
 - effets du traitement 7
 - remarques relatives aux performances 92
- SQL statique
 - effets du traitement 7
 - performances 92
- SQL0965 (code d'erreur) 134
- SQL0969 (code d'erreur) 134
- SQL1338 (code d'erreur) 34, 134
- SQL30020 (code d'erreur) 134
- SQL30060 (code d'erreur) 134
- SQL30061 (code d'erreur) 134
- SQL30073 (code d'erreur) 134
- SQL30081N (code d'erreur) 134
- SQL30082 (code d'erreur) 134
- SQL5043N (code d'erreur) 134
- SQLCODE
 - fichier de mappage 67
 - mappage 67
 - zone de la structure SQLCA 124
- SQLDA (SQL descriptor area)
 - taille allouée 92
- SQLSTATE
 - codes de classe 67
- structure SQLCA (SQL Communication Area)
 - mémoires tampons de données 124
 - SQLCODE (zone) 124
- support CCSID bidirectionnel
 - BIDI (paramètre) 35
- Sysplex
 - configuration requise 105
 - équilibre de la charge 105
 - informations sur les niveaux de priorité 105
 - paramètre 35
 - remarques liées à zSeries 104
 - support DB2 Connect 103
 - tolérance aux pannes 105
 - utilisation 105

T

- taille de bloc 107
- taille du bloc de pagination 107
- TCP/IP
 - ACCSEC (commande) 126
 - DOMAIN 34
 - extensions RFC-1323
 - changement d'échelle des fenêtres 117
 - noms d'hôte 40
 - noms d'hôtes éloignés 34, 40
 - noms de service 34
 - numéros de ports 40
 - port utilisé pour la resynchronisation 34
 - RESPORT 34
 - SECCHK (commande) 126
 - sécurité
 - scénarios 54
 - vérification 53
 - TCP/PORT 34

- TCP_KEEPALIVE
 - paramètre de configuration du système d'exploitation 86
- temps de réponse 89
- test
 - mises à jour multisites 63
- tests de performances
 - performances 89
- trace facility
 - traces DRDA 128, 133
- trace utility 124
- traces
 - données entre DB2 Connect et le serveur 124
 - fichier de sortie 124, 125
- transactions
 - applications réparties XA 65
 - DB2 Connect Enterprise Edition 27
 - débit 89
 - mises à jour multisites 11, 61
 - moniteurs de traitement des transactions 27
 - réparties
 - serveurs pris en charge 61
 - support 65
 - unité d'oeuvre (UOW) 11
 - validation en deux phases 11
- transfert de données
 - entre un hôte et un poste de travail 141
- tutoriels
 - identification et résolution d'incidents 153
 - Visual Explain 153
- Tuxedo
 - DB2 Connect Enterprise Edition 27
- type de données en virgule flottante 118
- type de sécurité PROGRAM 54
- types de données
 - CHAR 118
 - conversion
 - effet sur les performances 118
 - décimaux non condensés 118
 - données de type caractère 118
 - format décimal condensé 118
 - INTEGER 118
 - VARCHAR 118
 - virgule flottante 118

U

- unité d'oeuvre éloignée
 - caractéristiques 13
 - exemple 13
 - présentation 13
- unité d'oeuvre répartie
 - caractéristiques 11
 - mises à jour multisites 61
 - serveurs pris en charge 61
 - validation en deux phases 61
- unités d'oeuvre (UOW)
 - à distance 13
 - définition 11
 - réparties 61
- utilitaire d'administration
 - DB2 Connect 8
- utilitaire d'exportation
 - transfert de données entre un hôte et un poste de travail 141
- utilitaire d'importation
 - transfert de données entre un hôte et un poste de travail 141

- utilitaire de contrôle de l'état des processus 122, 126
- utilitaire ps (process status) 122, 126
- utilitaires
 - administration, DB2 Connect 8
 - db2drdat 124
 - ddcspkgn 57
 - définition des accès 57
 - état des processus 126
 - moniteur du gestionnaire de bases de données 8
 - ps (process status), état des processus 122, 126
 - trace 124

V

- valeur d'authentification 33
- valeur du paramètre VALIDATE RUN 126
- validation en deux phases
 - activation 61
 - port de resynchronisation utilisé par les connexions TCP/IP 34
- VARCHAR (type de données)
 - description 118
- variable de registre DB2CONNECT_IN_APP_PROCESS 73, 95, 104
- variables de registre
 - db2_connretries_interval 84
 - db2_max_client_connretries 84
- Virtual Telecommunications Access Method (VTAM) 54
- Visual Explain
 - tutoriel 153
- VM
 - DRDA
 - avec DB2 Connect 12
- VSE, DRDA 12

W

- WebSphere
 - Advanced Edition 21
 - Enterprise Edition 21
 - fonctions 21
 - présentation 21
 - Standard Edition 21
- Windows
 - moniteur de performances 74

X

- XA
 - connexions sécurisées 47

Z

- z/OS
 - DRDA 12

Comment prendre contact avec IBM

Pour contacter IBM dans votre pays ou votre région, reportez-vous à l'annuaire en ligne des contacts internationaux d'IBM à l'adresse suivante : <http://www.ibm.com/planetwide>

Pour en savoir plus sur les produits DB2, accédez à la page suivante <http://www.ibm.com/software/data/db2/>.



SC11-2427-00



Spine information:

IBM DB2 DB2 Connect version 9

Guide de l'utilisateur

